

DOCUMENT RESUME

ED 205 216

IR 009 523

AUTHOR Watson, Nancy A., Ed.
 TITLE Microcomputers in Education: Getting Started. Conference Proceedings (Tempe, Arizona, January 16-17, 1981).
 INSTITUTION Arizona State Univ., Tempe. Coll. of Education.
 PUB DATE Jan 81
 NOTE 349p.; Legibility varies.
 AVAILABLE FROM Gary Bitter, Arizona State University, Payne 203, Tempe, AZ 85281 (\$10.00).
 EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
 DESCRIPTORS Basic Skills: *Computer Assisted Instruction: *Computer Managed Instruction: *Computer Oriented Programs: Computer Programs: *Elementary Secondary Education: Evaluation Criteria: Federal Aid: Fine Arts: Gifted: *Microcomputers: School Administration: School Guidance: Special Education
 IDENTIFIERS BASIC Programing Language: *Computer Literacy

ABSTRACT

Included in these proceedings are brief write-ups of many of the 55 presentations given at a conference for elementary and secondary teachers and administrators. The strands of the conference emphasized using microcomputer technology in elementary education, secondary education, special education, and administration. General interest sessions were also held. The keynote address entitled "The Challenge of the 80's: Computer Literacy," was given by Dr. Andrew Molnar of the National Science Foundation. Sessions focused on: computer literacy, computer assisted instruction in the basic skill areas at elementary and secondary levels, applications for microcomputers in special education and gifted education, evaluating microcomputer systems, hardware and software comparisons, career education and guidance information systems, instructional techniques for teaching BASIC programing language to elementary and secondary students, designing computer proposals for federal funding, and microcomputers in the Fine Arts areas. Appended is a bibliography of BASIC computer books and lists of computer journals, film companies producing films about computers, microcomputer manufacturers, and software vendors. (Author/LLS)

 * Reproductions supplied by EDRS are the best that can be made
 * from the original document.

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.

Microcomputers in Education: Getting Started

CONFERENCE PROCEEDINGS

CONFERENCE DIRECTORS

Gary G. Bitter
Arizona State University

Patricia A. Walkington
Gilbert Public Schools

EDITOR

Nancy A. Watson
Arizona State University

Conference held at Arizona State University -
January 16th and 17th, 1981

"PERMISSION TO REPRODUCE THIS
MATERIAL IN MICROFICHE ONLY
HAS BEEN GRANTED BY

Gary G. Bitter

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)."

TABLE OF CONTENTS

INTRODUCTION

iv

Gary G. Bitter

I. GETTING STARTED	1.1
The Challenge of the 1980's: Computer Literacy.....	1.2
Andrew Molnar	
The Computer Literacy Curriculum	1.9
Mick Manning	
Computer-Based Systems for Career Based Information.....	1.14
Computer Literacy Using Microcomputers.....	1.18
Gary G. Bitter	
Curriculum Guide for Beginning Computer Literacy and Programming.....	1.19
in BASIC	
Patricia A. Walkington	
A Delivery System for Instructional Computer Resources.....	1.32
Robert Zimmerman and Pat Johnson	
The Making of a Genius Generation	1.37
Alan P. Hald	
Managing Instruction with the Micro	1.44
Aly Gundlach and Barbara Cirac	
Microcomputers in Education	1.55
Lorraine Mecca	
Microcomputers in Education - Funding.....	1.66
Henry Dahlberg	
Microcomputers in the Elementary Classroom	1.70
Polly Mack, Marilyn J. Latham and Milton Zuroff	
Microcomputer Management Systems	1.72
Samuel R. Romero	
MicroSIFT - A Clearinghouse for Microcomputing in Education	1.84
Donald C. Holznagel and Judith B. Edwards	
Network Systems for Schools	1.86
Cassie C. Connell	
One Thousand Students, One Computer - Problems!.....	1.95
Linn Boswell	
Programs Guaranteed Not to Teach.....	1.96
H. Donald Jacobs	

R _x For the 80's: Microcomputers and Problem Solving	1.102
Gary G. Bitter	
Strategies for Getting Started in Microcomputers	1.117
Patrick J. Carr	

II. COMPUTER ASSISTED INSTRUCTION

Business Education	2.2
C.A.I. in the Business Department	
Jeff Martin	
Microcomputers in Business Education	2.3
William Drum	
English	
A Middle School Language Arts Series	2.10
Robert L. Williams	
Gifted	
Computer Literacy	2.16
Fred Bell	
Computeronics.....	2.20
Pristin Bird	
Gifted Education...Enrichment or Acceleration?...Computers Provide.....	2.22
Both	
Marilyn Sue Ford, Patricia A. Walkington and Gary G. Bitter	
Instruction	
How to Design an Educational Microcomputer Program.....	2.32
Dan Isaacson	
Teaching Teachers to Teach Computing	2.60
Nancy Grimes	
Mathematics	
Arrowview Individualized Math	2.68
Craig Walker	
Microcomputer - A New Tune for an Old Song.....	2.79
Gena Trimble and Shirley Frye	
Music	
A Keyboard Based Digital Playing and Recording System.....	2.85
Charlie Kellner, Ellen Lapham and Laurie Spiegel	
Teaching Music with the Microcomputer.....	2.92
David Cohen	

Programming

BASIC for Elementary Students	2.94
Marilyn Sue Ford	
Language - An Alternative Approach to Programming.....	2.108
Eloise C. Babcock	
LOGO: A Computer Application for Elementary Children.....	2.120
John D'Angelo	
Personalizing Computer Programs to Your Needs.....	2.126
Pat Kuhn	

Science

Instructional Applications of Microcomputers in the Sciences.....	2.142
Daniel Davidson	
Using a Microcomputer in Chemistry and Physics.....	2.147
John Friend	

Special Education

Blooming with the Micro.....	2.151
Peggy Comer	
Microcomputer Assisted Instruction with Learning Disabled Students.....	2.164
Marley Watkins	
A New Approach to the Same Old Problems.....	2.173
Miriam Furst	
Using Microcomputers with Blind and Deaf Children.....	2.179
Roger Goodberlet	

III. GENERAL INFORMATION.....	3.1
BASIC Computer Books.....	3.2
Computer Journals.....	3.4
Computer Related Films.....	3.5
Microcomputer Manufacturers.....	3.7
Software Vendors.....	3.8

INTRODUCTION

The Microcomputers in Education: Getting Started Conference was held at Arizona State University, Tempe, Arizona on January 16, 17, 1981. The conference was sponsored by the College of Education. There were over 400 participants in attendance with 55 speakers and 18 exhibitors. The conference was for Elementary and Secondary teachers and administrators. The strands emphasized elementary education, secondary education, special education, administration and general interest sessions. Dr. Andrew Molnar of the Natural Science Foundation gave the keynote address entitled "The Challenge of the 80's: Computer Literacy." Dr. Judy Edwards of the Northwest Regional Educational Laboratory was the featured luncheon speaker presenting the topic "The Courseware Crisis: Help is on the Way." The sessions were on Computer Literacy; Computer Assisted Instruction in the Basic Skill areas at Elementary and Secondary Levels; Applications for Microcomputers in Special Education and Gifted Education; Evaluating Microcomputer Systems; Hardware and Software Comparisons; Career Education and Guidance Information Systems; Instructional Techniques for Teaching Basic Programming Language to Elementary and Secondary Students; Designing Computer Proposals for Federal Funding; and Microcomputers in the Fine Arts Areas.

The proceedings represents a brief write-up of many of the sessions. Each speaker was asked to write a three or more page write-up on their talk. What follows are the papers which the speakers submitted. Each paper represents the feelings and detail of the speaker. No attempt was made to change any paper.

In conclusion our intent is to bring to the reader a quick review of microcomputers In Education as presented to the participants of the Ninth Annual Math/Science Conference at Arizona State University.

Gary G. Bitter
Conference Director

II. GETTING STARTED

THE CHALLENGE OF THE 1980s: COMPUTER LITERACY

By
Dr. Andrew Molnar
Project Manager
Research in Science Education
National Science Foundation
Washington, D.C. 20550

Today more than half of our GNP and half of our jobs are in the information industry. Computers are the basis of our industry. This change from an industrial to an information society is creating a crisis for educators. We must ask, "Are we preparing our children for the world in which they live?"

The United States uses advanced technology more than any other country in the world. The Controller General says that the greatest hope for increasing our rate of production and our growth lies in technological innovation. He says our national policy for full employment requires technological innovation on an order not achieved in the past decade. The United States for the past two decades has been the finder, user and forger of high technology.

However, other countries are now gaining on us. West Germany and Japan are going to offer clear competition for us in the near future. Foreign firms are not only working cooperatively with commercial firms but with their educational institutions as well. They are working to ensure that computing is introduced into the schools at a very early age.

Last October France's President Valery Giscard d'Estaing declared a special week for "computers in society" to emphasize the importance of computers in the French economy. The French are installing 10,000 microcomputers in their high schools throughout the country as part of their 10-year plan to make their country high in computer literacy.

The National Science Foundation (NSF) has conducted an assessment of science in the U.S. Clearly there is a decline in the offering of science courses in our school systems. In high schools there is an increasing amount of science illiteracy. The report says this is attributable to declining enrollments, increasing costs and relatively ineffective support programs.

Education is 7.9% of the Gross National Product (GNP). This surpasses health at 6.6% and defense at 5.5%. In fact, the percentage of the GNP for education has exceeded defense since 1973. However, we have not provided support for our science and mathematics teachers. This is due in part to strong public dissatisfaction with the innovations of the 1960s. Many people feel that instead of introducing computers into the curriculum we should return to teaching basic skills. However, today we must ask, "What is basic?"

According to an NSF report entitled "Science, Engineering and Education for the 1980s and Beyond," we as a nation are becoming technologically illiterate. There is a growing discrepancy between the kind of education high school students who want to pursue a career in science or math are receiving and the technology of the future. Those who do not major in the sciences only have very rudimentary notions of science, math and technology, and this may become a very serious problem in the next decade. Only one-third of the nation's 1700 school districts require graduates to take a year of math and science for graduation.

A study conducted at the Math Department of the University of Chicago concludes that the Soviet population is getting an education that only a very few of our best students are receiving. In the Soviet Union, 97% graduate from secondary school compared to 75% in the United States. The

Soviet citizen gets 13 years of education compared to 12 in the United States. All graduates from a Soviet secondary school will have 2 years of calculus as opposed to 105,000 students in the United States who receive one year of calculus.

The Soviet student's science curriculum is also very intensive, including 5 years of physics, 4 years of chemistry, 5.5 years of biology, 5 years of geography and one year of astronomy. On the other hand, in the United States only 9.1% of our students graduate with one year of physics, 16% with one year of chemistry, 45% with one year of biology and 17% with one year of general science. It is worthwhile to point out that the Soviet Union is making an intensive effort to educate its entire population in science and technology.

The question before us in the next decade is "what are the capabilities of a computer?" Dr. Seymour Papert of MIT, working with very young children, initially took what he called the "turtle," a computer-driven device, and created a whole curriculum around the turtle. The children learned very well in this new computer environment. The interesting thing is that we have not challenged our children young enough. We can with problem-solving curriculum. The computer permits us to leapfrog over many skills that children don't already possess. For example, children cannot play musical instruments, therefore they are incapable of writing music. The computer permits children to write music, even exceptional compositions, easily.

In the future, a lack of knowledge of computers will make people as functionally illiterate as the inability to read, write or do arithmetic is today. Computers are involved in all aspects of our society. I would argue that computer literacy is the responsibility of the individual. What happens to the American engineer who graduates with honors but who has not been

exposed to computers? Can he work in industry? Imagine the man who followed a vocational program to learn how to run a drill press only to find out that you do not run drill presses any more. Instead you write a program to run a whole battery of drill presses. Imagine the secretary who has just learned touch typing only to find out she has to use a word processor.

How many college educated people understand the inner workings of the technological items they use every day? This failure comes from an ignorance of technology. We are subjects rather than masters of technology. One of the things that undermines self confidence is that never have there been so many people who are so dependent on a tool that they can neither understand nor repair. Our goal should be to restore human understanding and confidence in technology and to liberate the individual.

An interesting event occurred in England last April. A man on April Fools Day started the rumor that they were going to make Big Ben digital. Needless to say, the public became upset. This points out the fear and anxiety of technology. Like most people, some teachers fear technology. They are afraid they are going to be replaced by computers. One teacher in a mathematics journal is quoted as saying, "If I use digital watches, kids won't know how to tell time any more."

On the other hand, I think we all appreciate computer technology to a certain extent. The federal government is probably the largest user of computers and this is beneficial to us because it increases efficiency and productivity.

Everyone is affected by computers. In a consumer survey it showed that in one year about 40% of those surveyed found an error in their bill, and 15% found two such errors. Seven percent gave up trying to resolve these

errors, 8% never even tried to resolve the errors and 16% felt that in some way they were being coerced into paying. Without understanding how billing works, the common response is that a computer did it and that computers do not make mistakes.

The National Council of Mathematics Teachers says the most important thing in the next decade is problem solving in the schools. They also say that all students should have access to computers through their school's mathematics programs. Computers should be integrated into the curriculum. Courses in computer literacy should also be offered to teachers and be a requirement for certification.

Are we ready for an introduction to computers? Are students interested? A national assessment study finds that students say that they would like very much to have computers in their math classes and think they would be very useful. The NSF supported a study called "Priorities in School Mathematics." The idea was to go out and talk to teachers, parents and students before any curriculum revisions were made and find out what they thought was important. The interesting finding in this study is that lay samples gave stronger support for the use of computers than did professionals.

Are children capable of working with computers? Thirteen-year-old students are already reportedly breaking into sophisticated systems that had some safeguards. Dr. Papert has been working with children using microcomputers. He said that the student in the lower level of an average third grade class ought to be able to do something that he feels is interesting with the computer within ten minutes. He should then be able to move from there to conceive, develop, write and debug his own programs. Teachers without a previous knowledge of programming should be able to learn after about a half year of in-service training to write programs that will

change the interface between the user and the system. Dr. Papert is now doing a project in the Dallas area where third graders are going to be totally saturated with computers in the classroom and in their homes.

Teachers and educators who received their degrees BC (before computers) are already being challenged with technology people. The country as a whole will benefit from access to information that was formerly available to only a very small group of people. The microcomputer liberated us from the centralized control computer. You can now buy your own computer. Microcomputers are now feasible for all schools and most classrooms. The immediate need is to make certain that every high school graduate has an understanding of the uses and applications of computers in society and the impact they have on our daily lives.

We must permit students to use computers as they would typically be used in business, science and industry. We don't want to have courses that do not permit us to gain the confidence we will so sorely need in the coming decade.

We must increase the quality of education by introducing computer-based curriculum. We must review unilaterally all our curriculum and ask what is important and why, and create an environment in which our students will learn. The priority is becoming increasingly clear to both our national needs for increased productivity and our local needs for educational change. Greater cooperation is required by all segments of our society if we are to meet these needs. If we do not revitalize our educational institutions to meet this challenge, we are liable to become the next failing industry following in the path of the railroad industry.

Our society is growing more and more complex. The information explosion has created a discontinuity in the nature of our educational needs. Science curricula designed to meet these needs have been criticized as being too difficult and too complex for most students. Many feel that we should reduce the amount of science taught and instead teach more basic skills. However, the computer is ideal for dealing with complexity, and difficulty is not inherent in the subject matter but in our ability to cope with it. The computer provides us with a problem-solving tool capable of overcoming these difficulties. We should ask what are basics in an informational society, and move ahead not back to basics.

Reprinted with permission from Educational Computer. February 1981.

M. Dundee Maples, Publisher
Joan R. Hiraki, Editor
P.O. Box 535
Cupertino, CA 95015

Subscription rate - \$12.00 per year
\$20.00 first class to Canada and Mexico
\$26.00 airmail to all other countries

THE COMPUTER LITERACY CURRICULUM
PRESCOTT UNIFIED SCHOOL DISTRICT #1

MICK MANNING
COMPUTER LITERACY INSTRUCTOR - JUNIOR HIGH LEVEL
146 S. Granite
Prescott, Arizona 86301

Our Computer Literacy Program has three points of emphasis: 1) Careers, 2) programming, and 3) the social impact of computers on society. Limited by a 9-week program, it is impossible to cover everything thoroughly, but we do emphasize the high points.

The program is required of all 7th graders as part of the "mini course" program. Many of the students have received previous computer experience in 4th, 5th, and 6th grade, so the typical 7th grader will have had 18 weeks of "hands-on" computer experience by the time they leave 7th grade.

The course is designed with as much flexibility as possible. The opportunity to visit a site where computers are in operation gives us a chance to see computer related jobs first hand. We explain to the students that programming and computer design are not the only way you can use computer training. When they see that automechanics, newspaper typesetters, and office secretaries also use computers on a daily basis, they gain increased awareness of the computers influence on their lives.

A discussion of the computer as it relates to careers is often a difficult subject for the average 7th grader to grasp. Many have plans for the future but cannot see the relationship between their dreams and the 16K machine in front of them. Field trips help draw the connections together, but class discussions are most effective. The discussions are generated by various questions. Does anyone's mom or dad work with a computer? How do you think a computer could help a (doctor, firefighter, plumber, etc.) do his or her job

better? Questions like these lead to discussions about computer applications to the world of work.

The final approach used to cover the careers topic is the current use of news articles. Rarely does a day go by where one of the news media does not talk about how, with the aid of a computer, something new has been discovered. Computer aided surgery, computer enhanced photography, computer created music and art, are all likely topics of magazine articles and news reports. We encourage the students to bring in articles that they find in newspapers and periodicals, and we discuss them in class. This tends to increase their awareness of these machines and how they may effect them now and in the future.

Our seventh grade Computer Literacy curriculum is approximately 50% programming in "BASIC". We justify this emphasis on programming with two arguments. First, we have over 60 computers in the school district representing a financial investment of \$100,000 in equipment and salaries, and we want to work that investment to death, and we do! There are very few idle moments for our machines. Secondly, we have found that the average Seventh grader is not interested solely in the social and career aspects of computer literacy. All they want is to get their hands on the machines. So we use programming as a lever to get them thinking about the total impact of the computer on their lives.

We know that all of our students will not be writing great programs after just nine weeks. For one thing, not everyone is cut out to be a programmer, any more than every football player can be a quarterback. The main objective of the programming section of the course is to give the kids a feeling for the problems that both the user and programmer must face in order to accomplish a

task. Programming gives them a real "hands-on" appreciation for the limitations of a computer, and the need to develop certain problem solving skills. These are talents that everyone needs, whether he or she becomes a programmer or not. One of the hardest things for them to realize is that a computer is not some electronic superman, able to leap all problems in a single bound. Like all other tools, computers have limitations. This idea of limitations is really brought home when we have the students play certain computer games. The students are surprised to discover that they can beat the computer at checkers, football, or backgammon, often answering more quickly and playing with greater creativity. The idea of beating a computer game really excites them, and when we list the program, they see that these games are only made up of statements and commands they already know.

Programming really helps them put things into perspective. They learn that "computer errors" are really programming or data entry errors. Also, they discover some of the problems they will run into if they place too much faith in the machine. We talk about the problem of depending on the computer for information. Programming with unreliable data can be a real problem, producing incorrect answers or information. All of this emphasizes one of the most critical computer literacy concepts, that of computer limitations and how to deal with them. Thus, we use a lot of programming in our curriculum, but the main objective is to lead the students to some of the more important computer literacy concepts.

Knowing that only a small percent of students will ever become programmers is not a major concern. We believe that the social and career aspects of computer literacy are reason enough to offer the course. The predictions for the 21st

century call for a work force where 90% of all jobs will be directly or indirectly related to computers. The social impact from this change (which is less than 20 years away) is enough to convince most people that computer literacy should join Reading, 'Riting and 'Rithmetic.

Many jobs will be eliminated with the introduction of robotics in the factory. Several Japanese companies are planning to have fully automated factories by 1985; many robots are already in the "field" now. This will eliminate many non-computer related jobs. This will create some of the most interesting problems as more and more people interact (if that is the right word) with androids for a large part of the day. Many people, not working in computer related fields will still use computers to transfer business transactions and mail, and a greater percentage will work at home using computerized communications to interact with their company or office.

The people who will suffer the most from changes in the work place will be high school graduates, who have little or no computer related skills. As educators, it becomes our responsibility to see that they are given an opportunity to develop these skills or we will be facing enormous social problems in the 90's. To add to the immediate problems, the latest predictions call for a serious teacher shortage by the late 1980's, as talented teachers in the arts and sciences leave teaching for more lucrative positions in the computer industry. This means that there won't be the personnel available to teach computer literacy at a time when the social impact will be greatest. We try and convey this feeling of urgency to the students so that they will be ready for the 90's.

We emphasize that the present computer revolution offers a mixed bag of both good and evil. One authority predicts that there will be more millionaires in the 80's and 90's due to the increased opportunities this revolution will offer. There will be even more free time as tedious jobs are taken over by process control devices. Robots will do our more dangerous jobs. And computerized production lines will increase our productivity and use dwindling natural resources more efficiently. At the same time, there will be increased dependence on these machines, greater opportunities for invasion of privacy, and high risks of computer fraud.

Half the battle with a computer literacy course is to make the students aware of the changes around us. They take this computer age for granted. They expect everything to be computerized and assume that this is good. We talk about the dangers as well as the benefits, hoping to prepare them for a future which may ask them if they want a particular portion of their lives computerized or not. We explain that someday they will be asked to vote on certain bills controlling computer use and abuse.

In closing, I would like to say that the main objective of the course is to encourage the kids to use the computer and to learn as much as they possibly can about it. We want them fully aware of its limitations and capabilities, and its social and career aspects of their future, so that they may actively deal with the inevitable changes of the next 2 decades - creatively and positively.



1535 West Jefferson, Room 249, Phoenix, Arizona 85007
Telephone (602) 271-3680

Arizona
State
Occupational
Information
Coordinating
Committee

No. 5

COMPUTER-BASED SYSTEMS FOR CAREER INFORMATION

Good career decision-making - from youth through retirement - requires accurate occupational and educational information. High unemployment, new occupations, dissatisfied college graduates and increasing numbers of women going to school and work are a few of the many reasons people need facts about jobs, schools and training opportunities.

Too often, those seeking such information have to search through numerous pamphlets, books and reports, only to find outdated statistics, generalizations, and "national averages." Arizona, along with many states, is installing computer-based systems of career information to provide people with the information they need.

These systems work!

Computer-based systems of career information are a proven, effective method for delivering occupational and educational information. People like to operate the computer terminals! Easy to understand programs put the user in control of one of modern society's most important pieces of equipment - the computer. This kind of delivery adds a motivational force of its own, encouraging use and reuse of the system. Studies show that people have fun using these systems, want to use them again, recommend them to peers, and talk about them with friends, teachers, counselors, parents and others.

Systems of career information have other benefits as well. Evaluation studies show that people gain occupational and education knowledge, receive new job ideas, and often change career plans after using these systems. A recent national study by Herbert Parnes highlighted the importance of occupational information, reporting that "...the extent of a youth's information about the world of work is positively linked to measures of his success in the labor market."¹

¹United States Department of Labor, Career Thresholds, Volume 6: A Longitudinal Study of the Educational and Labor Market Experience of Young Men. Herbert S. Parnes, project director, (Washington, D.C., 1977) p. 192.

A study by the College Board identified the need for occupational and educational information among the large number of adults in career transition, noting that "...they are most interested in information services, particularly specific information on jobs, careers, or educational opportunities".² But the best testimony to the value of these systems is the growing number of schools and social agencies using computer-based systems of career information as part of their counseling programs. Clients, counselors, teachers and administrators alike find these systems to be an invaluable resource for career decision-makers. They know that good career decisions are based on accurate information.

How Do They Work?

Designed as a tool for vocational and career counseling, the computers can, depending on input, respond to a series of commands such as: add; subtract; either/or; print; itemize; etc. They are programmed to turn to any one of six information files and relay information in any of the following categories: Occupations; Armed Service Occupations; 4-Year Colleges; 2-Year Colleges; Graduate Schools; and Financial Aid Information. Each of the six files contains a variety of subheadings which are full of pertinent information relative to the subject at hand. Designed to be selective for both positive and negative information, the computers punch out lists of careers and/or schools best suited to the individual as determined by the data fed into the machine.

During the Spring of 1980, information files on Arizona occupations and noncollegiate schools will be added to the system. The ASOICC recently received a federal grant from the National Occupational Information Coordinating Committee (NOICC) to expand the information files and the availability of these systems in Arizona.

Where Do I Go?

In Arizona, six computer sites connect with approximately 70 terminals, located primarily in high schools and community colleges. In addition, pilot programs are in Vocational Rehabilitation Services, a medium security prison and in a CETA program.

For access in a high school check with your counselor. Community colleges offer the system as a resource to the public in their career centers. Locations which provide public access to the system are listed on the next page with an address, phone number and contact person.

If you are a potential user of the computer-based system in Arizona you may obtain further information by contacting Chris VanPetten, Executive Director, ASOICC, 1535 West Jefferson, Room 345, Phoenix, Arizona 85007, or phone (602) 255-3680.

²Arbeiter, Soloman et al. 40 Million Americans in Career Transition: The Need for Information. (College Entrance Examination Board, New York, 1978.) p. 22.

Portions of this information sheet were reprinted from Computer-Based Systems for Career Information published by the Association of Computer-Based Systems for Career Information (ACSCI).

COMPUTER-BASED CAREER INFORMATION LOCATIONS

County	Location	Contact Person & Phone
Gila	Gila Pueblo College Six Shooter Canyon Globe, Arizona 85501 Available on a part time basis, call first.	Ted Roberts, Counselor (602) 425-7131
Maricopa	Glendale Community College Career Development Center Student Union, East End 6000 West Olive Glendale, Arizona 85302	Brad Harper, Coordinator (602) 934-2211 Ext. 332
	Mesa Community College Career Center 1833 West Southern, Portable Building 100 Mesa, Arizona 85202	Kathryn England, Career & Advising Technician (602) 833-1261 Ext. 367
	Phoenix College Career Center 1202 West Thomas Road Phoenix, Arizona 85013	Cliff Schweitzer, Career Program Advisor (602) 264-2492 Ext 607
	Rio Salado Community College Student Services Department 102 North 40th Street Phoenix, Arizona 85034	Joe Rodrigues, Placement Officer (602) 275-7881 Ext. 616
	Scottsdale Community College Student Services Bldg. 9000 East Chaparral Scottsdale, Arizona 85243	Mary Ellen McKeny, Career Planning and Placement (602) 941-0999 Ext. 278
Pima	University of Arizona Career Information Center College of Education, Room 244 Tucson, Arizona 85271 Fee Charged: \$5.00 first visit \$3.00 additional visits M-Th: 8 - 8 p.m. Fri.: 8 - 5 p.m.	Jean Clancy, Administrative Assistant (602) 626-4975
	Pima Community College - West Campus Student Center 2202 W. Anklam Tucson, Arizona 85709	Delfina Alvarez, Counselor (602) 884-6991

County	Location	Contact Person & Phone
	Pima Community College - Downtown Campus 50 W. Speedway Blvd. Tucson, Arizona 85705	Hans Klinkner Career Information Specialist (602) 884-6726
	Pima Community College - East Campus Education Center 7830 East Broadway Tucson, Arizona 85710	Shirley Howard, Career Technician (602) 884-6949
Santa Cruz	Nogales Public School Dist. 222 Plum, Room 210 Nogales, Arizona 85621	Mary Matz, Counselor (602) 287-4043
Yavapai	Yavapai College Career Center 1100 East Sheldon, Bldg. 1 Prescott, Arizona 86301	Bob Dresser, Vocational Counselor (602) 445-7300 Ext. 245
	Clarkdale Campus Black Hills Drive P.O. Box 248 Clarkdale, Arizona 86326	Pat Walsh, Counselor (602) 634-7501
Yuma	Arizona Western College Center for CHOICE Student Services Building P.O. Box 929 Yuma, Arizona 85634	Chuck DeMore, Career Admissions Advisor (602) 726-1000

Computer Literacy Needs

by Gary Bitter
Arizona State University
Dept. of Elementary Educ.
Tempe, AZ 85287

What do Prescott, Arizona, Topeka, Kansas and Scarsdale, New York have in common? Would you believe computer literacy projects? These schools are a few of the many in the United States that are involved in implementing computers into education. Prescott, Arizona's computer literacy project was the winner of the 1980 most innovative project in the United States awarded by Instructor magazine. This particular project involves fourth, fifth, and sixth graders in computer literacy learning. Generally the specific computer activity varies from school to school but essentially children are learning about computers.

Technology of microcomputer hardware is advancing at an exponential rate. Most of us have seen the development of the computer in our life time. This phenomena is providing children the opportunity to teach their parents about computers. Familiarization of the capabilities of the computer including its USES and MISUSES in our society is a must for children. Undoubtedly they will grow up in a world dominated by computers. In fact, few people in the United States are not already effected by computer activities. With this tremendous impact expected to continue in the years ahead, children must start now becoming knowledgeable about the computer.

The microcomputer is here to stay but are we educators ready for the thrust. Here are some of the needs that I see for the years ahead.

1. Computer Literate Teacher

First we must inform education leaders about microcomputers and their classroom applications. We need computer literate teachers to provide microcomputers experience for children. Remember Modern Math - Without teacher understanding and willingness to use microcomputers, computer literacy will not come about.

2. Microcomputer Curriculum Development

Education needs to develop curriculum ideas for educating children about microcomputers. Thought needs to be given to incorporate the microcomputer into not only the math curriculum but also Social Studies, Science, Reading, etc... Educators need to be involved with technologists to apply the microcomputer to a curriculum. We need a curriculum with computer literacy as an integrated part.

3. Microcomputer Software

Lets stop asking the question - "What microcomputer should I buy?" but rather find the software which meets the needs of the child in your classroom and then find a microcomputer to operate the software. Remember, hardware is much more advanced than software. Software needs to include lesson plans, assignment pages and diagnosis procedures. We just can't continue handing the child a diskette and expect great outcomes in learning. It takes planning and coordination with the curriculum. We need software libraries just as we have film libraries so teachers can share in the available software.

4. Technology Fears

Activities need to be developed to provide teachers, parents and the community opportunities to work with computers. Microcomputers in libraries, entertainment centers, universities, parks, stores, etc...need to be common place. Human access is needed to eliminate computer anxiety.

These are a few of the goals ahead for educators to pursue if we are to have computer literate children. Computer Literacy is not just going to happen - we are going to have to actively plan our strategies. Lets not keep making the same mistakes - the only way it will begin is with "Computer Literate teachers"!!

CURRICULUM GUIDE
FOR
BEGINNING COMPUTER LITERACY AND PROGRAMMING IN BASIC

BY
Patricia A. Walkington

An increase in computer technology has led to a need for educators to get children involved in computers. A recent article in the March issue of Newsweek titled, "The Classroom Computer" stated that by 1985 approximately 75% of all jobs will involve computers in one way or another and that people who are unfamiliar with their operations and uses will be at a disadvantage.

The purpose of the following curriculum guide is to provide educators with an outline, student objectives, and a list of materials and resources that can be used to teach middle and upper elementary students computer literacy and programming in BASIC. This guide was developed as a result of a computer summer school program, offered to middle and upper elementary grade students under the sponsorship of the Tempe (Arizona) Parents Gifted Association (T.P.G.) in 1980. It should be noted that the topics of computer literacy and programming in BASIC were taught simultaneously.

For further information contact:

Pat Walkington
Gilbert Elementary School
140 South Gilbert Road
Gilbert, Arizona 85234

CURRICULUM GUIDE FOR BEGINNING COMPUTER LITERACY AND PROGRAMMING IN BASIC

by
Pat Walkington
Gilbert Public Schools

Rationale: Computers are becoming increasingly important in our society. In education, they can be used to facilitate individualized instruction and provide flexibility to special programs such as the gifted or handicapped. In addition, they aid in creative activities such as problem solving and higher level thinking skills. More importantly, however, they assist educators in helping children prepare for the future, where one day computer literacy will be considered a basic skill.

Topic	Objectives	Materials and Resources
I. Computer Pre-test and pre-assessment	1. Students will be able to discuss, in writing, what they already know about computers and why they think it is necessary to learn about computers.	<u>Minnesota Computer Literacy and Awareness Assessment:</u> Minnesota Educational Computing Consortium, 2520 Broadway Drive, St. Paul, Minn. 55113.
II. What is a Computer?	1. The student will be able to identify the following computer terms: Algorithm, arithmetic unit, BASIC, bug, character, central processing unit, chip, control unit, data, debug, disk, flowchart, hardware, input, keypunch, lineprinter, loop, memory, microprocessor, output, program, programmer, punched card, soft-ware, string, terminal, time-sharing, etc.	"One Burger Coming Up." <u>Scholastic Newstime</u> , vol 50, April 27, 1978. Billings, Karen. <u>Are You Computer Literate?</u> Dilithium Press, P.O. Box 92, Forrest Grove, Ore 97116 Bitter, Gary. <u>Basic for Beginners.</u> McGraw/Hill, Inc., New York.
A. Computer terms		
B. What computers are used for		
C. What computers can do		
a) speed		
b) accuracy		
D. Computer Limitations		
a) cost		
b) soft-ware availability		
c) storage capacity	2. Students will collect articles from newspapers and magazines to share with classmates showing what computers can do and how they are used in our society. These collected activities will be used to develop a scrapbook.	Ball, Marion J. <u>Be A Computer Literate.</u> Creative Computing Press, P.O. Box 789-M, Morristown, NJ 07960 Rice, Jean. <u>My Friend The Computer.</u> T. S. Denison and Company, Inc., Minneapolis, Minnesota

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
II. What is a Computer? (continued...)	2. (continued...)	Rice, Jean. <u>My Friend The Computer: Teacher's Guide and Activity Book.</u> T. S. Denison and Company, Inc., Minneapolis, Minnesota
		Filmstrip: "Computers for Kids" by Donald Spencer, Camelot Publishing Co., P.O. Box 1357, Ormond Beach, FL 32074
		Movie: "What is a Computer?" Audio- visual services, Arizona State Univ.
	3. Students will be able to identify the following terms related to computer speed: millisecond, microsecond, nanosecond, picosecond.	Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
	4. Students will be able to discuss the limitations of the computer — such as its inability to synthesize infor- mation, and originate new ideas.	"One Burger Coming Up." <u>Scholastic Newstime.</u>
		Moursund, David. <u>Calculators, Com- puters and Elementary Education.</u> Dept. of Computer Science, Univ. of Oregon, Eugene, Oregon 97403.
		Spencer, Donald D. <u>Computers for Kids.</u> Camelot Publishing Company, P.O. Box 1357, Ormond Beach, FL 32074.
III. History of the Computer	1. Students will be able to describe why the computer might have been invented (to help man extend the use of his brain and keep track of large numbers of things).	Moursund, David. <u>Calculators, Com- puters and Elementary Education.</u>
A. Why was the com- puter invented?		Rice, Jean. <u>My Friend the Computer.</u>
B. Abacus		
C. Blaise Pascal-1642		
D. Jacquard-1800		
E. Babbage-1820	2. Students will be able to discuss the inventors who contributed to the de- velopment of the computer.	Spencer, Donald D. <u>The Story of Computers.</u> Camelot Publishing Co., P.O. Box 1357, Ormond Beach, FL 32074.
F. Herman Hellerith- 1890		
G. Mark I		

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
III. History of the Computer (continued...) H. ENIAC-1946 (1st Generation of Computers) I. Second Generation (Transistors) J. Third Generation (Integrated Circuits) K. Fourth Generation (Microchip)	2. (continued...) 3. Students will be able to identify the four generations of computers.	Rice, Jean. <u>My Friend the Computer.</u> Billings, Karen. <u>Are You Computer Literate?</u> Rice, Jean. <u>My Friend the Computer.</u> Moursund, David D. <u>Calculators, Computers and Elementary Education.</u> "The Science Fair Story of Electronics" by Radio Shack, Fall 1979. Micro-Shopper: <u>The Microcomputer Guide.</u> Microage, 1425 W. 12th Pl., Tempe, Ariz 85281. <u>Classroom Activities and Reference Guide for Use with the Computer Parts Kit.</u> The Educational Computer Shoppe, Rt 4, Box 5, Cambridge, MN 55008. <u>The Computer Parts Kit.</u> The Educational Computer Shoppe, Rt. 4, Box 5, Cambridge, MN, 55008.
IV. How Computers Work A. Analog and Digital Computers B. Computer Parts a) input b) output C.P.U. c) arithmetic unit d) memory unit e) control unit	1. Students will be able to compare the basic functions of an analog computer and a digital computer. 2. Students will be able to describe, in general, the parts of a computer and how they work together.	Spencer, Donald D. <u>Computers for Kids.</u> Rice, Jean. <u>My Friend the Computer.</u> Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u> Ball, Marion J. <u>Be a Computer Literate.</u> Rice, Jean. <u>My Friend the Computer.</u>

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
IV. How Computers Work (continued...)	2. (continued...)	Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
	3. Students will be able to list things which could be compared to a compu- ter because they use input, processing, and output,	Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
	4. The student will be able to describe two different types of information that can be stored in the computer's memory. (Data, Program)	Rice, Jean. <u>My Friend the Computer.</u> Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
V. Getting Information Into and Out of a Computer	1. Students will be able to differentiate between the following input devices: terminal, cathode ray tube (CRT), punched card, and paper tape reader.	<u>The Computer Parts Kit.</u> <u>Classroom Activities and Reference Guide for Use with the Computer Parts Kit.</u>
A. Input Devices		
B. Output Devices		
C. Storage Devices		
D. Storage Capacity		
a) speed		Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
b) size		
	2. Students will be able to identify the code on a punched card.	<u>Classroom Activities and Reference Guide for use with the Computer Parts Kit.</u> Rice, Jean. <u>My Friend the Computer.</u>
	3. Given a punched card, students will be able to develop a coded message.	Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>
	4. Students will be able to differentiate between the following output devices: terminal, CRT, line printer, paper.	<u>The Computer Parts Kit.</u> <u>Classroom Activities and Reference Guide for use with Computer Parts Kit.</u> Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u>

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
V. Getting Information Into and Out of a Computer (continued...)	5. Students will be able to identify the following types of storage devices: Magnetic tape, magnetic disk, magnetic core. 6. Students will be able to list the advantages and disadvantages of the above storage devices with regard to size of storage and operating speed.	Classroom Activities and Reference Guide for Use with Computer Parts Kit. Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u> <u>Classroom Activities and Reference Guide for use with Computer Parts Kit.</u>
VI. Types of Computers		
A. Time Sharing		
a) terminal	1. Students will be able to describe the time-sharing system of a large complex computer, and the process of being on-line.	Rice, Jean. <u>My Friend The Computer.</u>
b) on-line		Rice, Jean. <u>My Friend The Computer: Teacher Guide and Activity Book.</u>
c) advantages		
d) disadvantages	2. Students will be able to demonstrate the process of logging-on to the computer if a time-sharing system is used.	Field trip: TIOCIC : Phoenix College, 15th Avenue and Thomas Road, Phoenix, Arizona.
B. Mini-computers		
a) uses		
C. Microcomputers	3. Students will be able to discuss the advantages and disadvantages of time-sharing computer systems.	Rice, Jean. <u>My Friend The Computer.</u>
a) uses		
b) parts	4. Students will be able to discuss the uses of mini-computers.	
c) advantages		
d) limitations	5. Students will be able to list the uses of microcomputers in the home, business and education.	Waite, Mitchell. <u>Microcomputer Primer.</u> Indiana, Howard W. Sams & Co., 1980.
		Friedman, Arnold, <u>Computer Programming Basic for Microcomputers.</u> Educational Activities, Inc., P.O. Box 392, Freeport, N.Y., 11520.

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VI. Types of Computers (continued...)	5. (continued...)	(Use a local vendor, such as Radio Shack, to give a demonstration on microcomputers.)
	6. Students will be able to identify the following microcomputer parts and terms: input, output, CPU, arithmetic unit, memory, (ROM, RAM), storage devices (magnetic tape, magnetic disk, floppy disk), micro-processor, simple binary information (bit), byte, chip, CRT, etc.	<u>Friedman, Arnold. Computer Programming Basic for Microcomputers.</u>
	7. Students will be able to discuss the advantages and limitations of the different types of microcomputers.	Field trip to: Computerland, or Microage.
VII. Communicating Instructions to the Computer A. Flowcharting B. Programming C. Different Computer Languages a) BASIC b) Fortran c) COBOL D. Software and Hardware	1. Students will be able to explain why flowcharts are helpful in developing computer programs.	Albrecht, Robert L. <u>Teach Yourself Basic</u> . Technical Educational Corp., 1970. "One Burger Coming Up." <u>Scholastic Newstime</u> . Ball, Marion J. <u>Be a Computer Literate</u> .
	2. Students will be able to identify the symbols used to develop a flow-chart.	Rice, Jean. <u>My Friend the Computer</u> . Albrecht, Robert L. <u>Teach Yourself Basic</u> . Bitter, Gary. <u>Basic for Beginners</u> . Rice, Jean. <u>My Friend The Computer: Teacher's Guide and Activity Book</u> .

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VII. Communicating Instructions to the Computer (continued...)	3. Students will be able to develop a flowchart for an activity that they do daily (mowing the lawn, getting up in the morning, doing homework, etc.). 4. Students will be able to describe the steps involved in solving a problem with a computer. 5. Students will be able to differentiate between several different types of computer languages, including BASIC, Fortran, COBOL, etc. 6. Students will be able to differentiate between software and hardware.	"One Burger Coming Up." <u>Scholastic Newstime</u>. Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book</u>. Ball, Marion J. <u>Be A Computer Literate</u>. Rice, Jean. <u>My Friend the Computer</u>. Ball, Marion J. <u>Be a Computer Literate</u>. Friedman, Arnold, <u>Computer Programming: Basic for Microcomputers</u>. Filmstrip IV and V. Spencer, Donald D. <u>Computers for Kids</u>
VIII. Programming in BASIC	1. Using a Cardboard Illustrative Aid to Computation (CARDIAC), students will be able to describe the operation of a computer, and compute simple problems including addition, multiplication, looping, and reversing the order of numbers.	Hagelbarger, David. <u>A Cardboard Illustrative Aid to Computation</u> . Bell Telephone Laboratories, Inc., 1969.
A. Computer operation through illustration B. Characters C. Variables D. Constants E. Basic Operations F. Basic Expression Evaluation		

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VIII. Programming in BASIC (continued...) G. Scientific Notation H. BASIC statements I. Relation symbols J. BASIC Commands K. Library Functions L. Arrays	2. Students will be able to identify the characters, letters of the alphabet, ten decimal digits, and special characters (+, =, /, etc.) that are used in BASIC programming language. 3. Students will be able to differentiate between constants (numbers), and variables (letters, and letter-number combinations), and be able to use both correctly in developing simple programs. 4. Students will be able to identify the mathematical operations used in the BASIC programming language (+, -, *, /, ↑, (),). 5. Students will be able to identify the rules used by the computer to determine the order of evaluation of a mathematical formula. 6. Students will be able to explain how the computer uses scientific notation when dealing with very large or very small numbers. 7. Students will be able to identify the functions of the following BASIC	<u>Visual Masters for Teaching BASIC Programming.</u> Camelot Publishing Co., P.O. Box 1357, Ormond Beach, FL 32074. <u>Visual Masters for Teaching BASIC Programming.</u> Bitter, Gary. <u>Basic for Beginners.</u> <u>Visual Masters for Teaching BASIC Programming.</u> Rice, Jean. <u>My Friend the Computer: Teacher's Guide and Activity Book.</u> <u>Computer Programming, BASIC for Microcomputers: Filmstrip II.</u> Bitter, Gary. <u>Basic for Beginners.</u> <u>Visual Masters for Teaching Basic Programming.</u> Radio Shack. <u>Basic Computer Language is Easier Than You Think.</u> Texas, Tandy Corporation, 1977. Coan, James S., <u>BASIC, BASIC: An Introduction to Computer Programming in Basic Language.</u> New Jersey, Hayden Book Company, Inc. <u>Visual Masters for Teaching Basic Programming.</u>

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VIII. Programming in BASIC (continued...)	7. (continued...) statements: LET, PRINT, INPUT, READ/ DATA, RESTORE, GOTO, STOP, END, GOSUB, DIM. 8. Students will be able to identify and define the following relation symbols: = ≥ ≤ < > 9. Students will be able to distinguish between conditional branching using the IF/THEN statement, and un-conditional branching using the GOTO statement. 10. Students will be able to describe how looping and nested loops are accomplished using the FOR/NEXT statement and will be able to develop programs using loops and nested loops.	Bitter, Gary, <u>Basic for Beginners.</u> Albrecht, Robert L. <u>Teach Yourself Basic.</u> Coan, James S. <u>BASIC, BASIC: An Introduction to Computer Programming in Basic Language.</u> Albrecht, Robert L. <u>My Computer Likes Me When I Speak in Basic.</u> Dymax, P.O. Box 310, Menlo Park, CA 94025 <u>Visual Masters for Teaching Basic Programming.</u> Bitter, Gary. <u>Basic for Beginners.</u> <u>Visual Masters for Teaching Basic Programming.</u> Albrecht, Robert L. <u>Teach Yourself Basic.</u> Friedman, Arnold. <u>Computer Programming: Basic for Microcomputers.</u> Filmstrip III. Bitter, Gary. <u>Basic for Beginners.</u> Albrecht, Robert L. <u>Teach Yourself Basic.</u> <u>Visual Masters for Teaching Basic Programming.</u>

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VIII. Programming in BASIC (continued...)	11. Students will be able to identify and use the following commands in the development of their computer programs: RUN, LIST, CATALOG, RANDOMIZE, REM. 12. Students will be able to identify and use the following library functions in the development of their computer programs: RND, INT, SQR, TAB. 13. Students will be able to develop a program using a single-subscripted variable in a one-dimensional array. 14. Given sample programs, students will be able to analyze the programs, and determine the value of each variable to be printed out prior to running the program on the computer.	<u>Visual Masters for Teaching Basic Programming.</u> Bitter, Gary. <u>Basic for Beginners.</u> <u>Visual Masters for Teaching Basic Programming.</u> Spencer, Donald D. <u>Game Playing with Basic.</u> New Jersey, Hayden Book Company, 1977. Peckham, Herbert D. <u>BASIC A Hands on Method.</u> McGraw Hill, Inc., 1978. Albrecht, Bob. <u>BASIC for Home Computers - A Self Teaching Guide.</u> Dymax, P.O. Box 310, Menlo Park, Calif. 94025. Peckham, Herbert D. <u>BASIC A Hands On Method.</u> <u>Visual Masters for Teaching Basic Programming.</u> Bitter, Gary. <u>Basic for Beginners.</u> <u>Visual Masters for Teaching Basic Programming.</u> Coan, James S. <u>BASIC BASIC: An Introduction to Computer Programming.</u> Peckham, Herbert D. <u>BASIC A Hands On Method.</u>

Curriculum Guide for Beginning Computer
Literacy and Programming in Basic

Topic	Objectives	Materials and Resources
VIII. Programming in BASIC (continued...)	15. Given a problem, students will be able to use combinations of the above statements, commands, and functions to develop a computer program to solve the problem. 16. Students will be able to design and develop an original computer program.	Weibe, James. <u>Basic Programming for Gifted Elementary Students.</u>
IX. Computer Advantages and Disadvantages A. Advantages: a) used in all areas of our society b) store and retrieve information c) are accurate and fast d) create new jobs e) assist in education B. Disadvantages: a) used to commit crimes b) may cause an invasion of privacy c) decreases employment d) lack of software development	1. Students will be able to discuss how and why computers are used in the areas of medicine, transportation, business, law enforcement, industry, education, communication, aerospace, etc. 2. In addition to discussing the misuses of computers in our society, students will be able to brainstorm possible future solutions to these problems.	"One Burger Coming Up." <u>Scholastic Newstime.</u> Billings, Karen. <u>Are You Computer Literate?</u> Ball, Marion J. <u>Be a Computer Literate.</u> Rice, Jean. <u>My Friend the Computer.</u> Have students refer to the newspaper and magazine clippings they have been collecting for their scrapbooks.
X. Computer Careers	1. Students will select a computer career that they are interested in, and collect information through an interview process, to share with other students in their class.	Resource people and guest speakers in computer careers.

1.31

43

A DELIVERY SYSTEM FOR INSTRUCTIONAL COMPUTER RESOURCES

Ms. Pat Johnson
COMP-YOU-TERS Project
Watertown Senior High School
Watertown, South Dakota 57201

Dr. Robert Zimmerman
Director of Computer Services
Northern State College
Aberdeen, South Dakota 57401

First Instructional Use of Computers. In the late 1950's and early 1960's, computers made their first thrust into the educational marketplace. Great plans were made for the ultimate teacher, a computer, that could individualize instruction for every student. In fact, many well-known journals such as The Mathematics Teacher in March, 1958, were printing articles like, "A High School Digital Computer Program: How Much or How Many?"

A collage of headlines from prominent periodicals and journals yields some insight to the initial enthusiasm that accompanied computers first shot at educational applications. American Business Education presented "The Computer As A Tool In Education" in May, 1962. "Technology is Knocking at the Schoolhouse Door" was written by Charles Silberman for Fortune in August, 1966. Patrick Suppes wrote about "The Uses of Computers In Education" in the September, 1966 issue of Scientific American. Many other articles had appeared for about a decade when in May, 1967 Saturday Review published, "Kids, Computers and Corporations" in which it stated, "There has been increasing concern that students will become appendages to the electronic monsters".

One should note that the appearance of so many articles that were "beating the drums" for computer use in education up to twenty years

ahead of the capacity to deliver a viable, cost-effective product represents a classic case of overpromising. Change, even under favorable circumstances, is a slow process that is encumbered with many pitfalls. In the case of computers in education, hardware was expensive and unreliable, software or courseware had not been developed, and trained personnel were not available in the schools. Small wonder that the use of computers in education didn't catch on!

In retrospect, it is easy and somewhat amusing to review the journal articles. We can see now that the ideas proposed for use of computers were sound. Many are now the basis for functioning programs in schools throughout the United States. Within another five years many more schools will, no doubt, be added to the list of users. We have emerged from the decade during the late sixties and early seventies when there was a good deal of reluctance to try computers in school settings.

COMP-YOU-TERS Project Develops. As we emerged from the period of "reluctance", the COMP-YOU-TERS Project was proposing a computer-based instruction project for South Dakota. At that time, less than five percent of the school districts of South Dakota were using any form of computer-based instruction. Seventeen school districts including twenty-five secondary schools joined together on a time-sharing network centered at Watertown Senior High School. The project was organized by Pat Johnson, who became the project director, and Bob Zimmerman, who became a consultant to the project.

During the first year of the COMP-YOU-TERS Project, equipment was selected for a time-sharing system; a PDP 11/34 minicomputer was ordered from Digital Equipment Corporation. Microcomputers were considered and it was decided to wait until the time-sharing system was in place to

evaluate the potential uses of micros.

In-service education became the focus of project activities. Coordinators were selected from each participating school and included in workshops. School presentations were made in each school. Teachers began to test first computer applications in the Spring of 1978.

With the time-sharing computer equipment functioning and in-service programs operating, attention was turned to courseware. Each year two curriculum areas were selected for courseware development. Mathematics, science, social science, and business subjects were the focus of development activities.

Microcomputers were introduced into the COMP-YOU-TERS Project during the second project year. Each year the micros played a larger role in providing computer resources for students. PET micros were selected by most schools with Apple II, Atari, and Radio Shack micros also used.

As the third year of project operation approaches its midpoint, over 8,000 students have been served. Every project school has adapted its curriculum to include computer-based instructional experiences for students.

Evaluation Findings. An evaluation design was formed to monitor project activities and to provide formative and summative evaluation. Data was collected and analyzed to determine the affective and cognitive outcomes with teachers and students.

Information available at the end of the second year of project operation indicated that about 54 percent of the secondary school students had used computer resources provided by the COMP-YOU-TERS Project and 30 percent had written programs. The difference between users and programmers was accounted for by students that only used

computer assisted instruction (CAI), ran simulation programs or "canned" programs, and/or accessed the Guidance Information System (GIS).

Students with high interest areas in science and mathematics expressed the greatest interest in learning about computers and vocational track students were least interested. Perhaps, this same statement could be found true for other subject areas. In support of this observation, it was also found that high achieving students enjoyed computer work more. Obviously, there was an interrelationship among these variables.

Programming experience had a favorable effect on cognitive achievement. Students with experience programming were more confident of their ability to work with computers and had a higher level of knowledge about computers than those students that had only used the computer or had no computer experiences. Students, regardless of their level of experience, felt that computer study was important. Sex of students had no significant effect on students interest in computers.

Teachers were surveyed each year to secure formative information for project management and to monitor project outcomes. Over 95 percent of the teachers felt that every student should learn about computers. Teachers rated student learning of social issues higher than learning of functional capability. That is, they rated learning about the potential uses of computers more highly than actually providing courses about computers.

Training of teachers in a 1980 survey, said their training about computers was not adequate. When asked if they had any form of computer training, math teachers (78%) and science teachers (71%) had the most training and English teachers (16%) and Social Science teachers (26%)

had the least training about computers. Teachers with the most training were most supportive of in-school use and most interested in additional computer coursework.

Summary. Educators took their first look at computers in the ten year period 1958-1968. Hardware, software, and courseware were not ready. Even with the enthusiasm for math/science during that period, the use of computers in schools was not successful.

Ten years passed and many improvements occurred in computer hardware. Software development and its integration into the school curriculum was undertaken. The COMP-YOU-TERS Project was organized in 1978 to put together the pieces for computer use in schools of South Dakota. Indications from the evaluation procedures used, show that the computer was a successful media of instruction in South Dakota. Schools joined together to coordinate their efforts and pooled their resources to make a place for time-shared and microcomputer resources in their classrooms.

GENIUS GENERATION

By Alan P. Hald
1425 E. 12th Place
Tempe, Arizona 85281

AN OVERVIEW

Less than three centuries ago, the invention of the "factory" as a system to produce goods - coupled with the economics of capitalism - created an era of incredible material growth and social change. Societies were catapulted into the future experiencing twenty-fold increases in per capital wealth - moving from agrarian based economies, to early industrial, to the mass consumption industrial economies of today.

In the eighties we are witnessing a parallel phenomena. The introduction of microelectronics into the home and office is creating the contemporary equivalent of the Industrial Revolution. Microelectronics technology is dramatically increasing our ability to recall information, communicate, create knowledge and understand complex relationships. We are moving from a society perceived as resource constrained to an information rich society.

As a measure of this change, by 1984 annual sales of personalized computers should surpass a million units. At the end of the decade, 20% of all American households and a far greater percentage of businesses, will own a computer; that represents 30 times the number of computers

in the world today. A similar phenomena will be occurring in urban centers throughout the world.

THE TECHNOLOGY

Today's advances in microelectronics has outpaced our ability to apply that capability to practical uses. The pace of miniaturization and cost reduction in computer power is astounding. 30,000 transistor like components can easily and cheaply fit on a chip of silicon roughly one-fourth the size of your fingernail. In several years, the limits should be pushed out to 100,000 components roughly equivalent to today's medium sized computer mainframes. Into the next decade, limits may reach over one-million components per chip with processors operating ten times faster than today. It's tough to imagine what we might do with the equivalent of 10 large computers compressed to the size of a wristwatch. Never-the-less, we can look forward to at least a thousand-fold increase in microcomputer capability by the early 1990's.

FRED THE HOUSE

Have you ever thought what it might be like to carry on an intelligent conversation with your house. Computers can already talk, interpret speech, and control appliances. Throw in a few sensors, a modest vocabulary, the telephone network and your house can talk to anyone or anything in the world. It is feasible within the next decade to build a home which can monitor its condition, maintain itself, and communicate to you and even other homes.

Imagine a house called Fred, who while performing routine roof maintenance discovers a leak. Fred first seeks help. Not from you, but

from Slim, a ranch-style home down the block, who has recently undergone roof repair and can provide Fred with needed advise. Fred then calls you at the office concerning his plan of action. You've learned to trust Fred's judgement and approve the repairs. The rest is rather straight forward. Fred calls the roofer and directs her to the leak. After it is repaired to Fred's satisfaction, funds are electronically transferred from your account to the roofers account. Fred promises to give her a good reference and stores the entire exprience in his memory banks for future use, to share with other homes and humans.

Think about growing up in a house like Fred. Slam the door or write on the walls and Fred will counsel you about it. You'd develop a different sensitivity to your environment, relating to it as if it were human. Extend this concept to your automobile or to a city. We'd become more aware of their needs through feedback provided by their own computer intelligence. As our technology becomes more self sustaining, our lives would become more convenient, possibly less boring - more adventuresome and self fulfilling. We'd be creating environments conscious of their own existence, aware of their interdependence with us - the human inhabitants.

At the National Office Machine Dealers Association (NOMDA) meeting two years ago, I presented a similar concept called Albert the Office. At that show we demonstrated that an office could talk - be talked to - control its environment - call up and converse with another computer automatically. The reactions of audiences exposed to both Fred and Albert are quite different. The concept of talking homes generally induces laughter but a talking office invokes a concerned silence. Implementation

of this technology in the 80's, clearly will introduce new social questions and choices, many of which may be perceived as threatening when applied to environments where we have little personal control.

ACCESS TO KNOWLEDGE

Within the next decade a typical office or home could have information equipment consisting of an affordable computer system, integrated with a large capacity information storage and recording devices such as video disks and recorders. The equipment would be tied into communication networks global in scope through the phone system, cable, television, and other broadcast media.

A small library could be quickly accessed through this equipment. For example, you could ask about zebras and you are shown a zebra running, you're told about its habitat and characteristics. The information is automatically adjusted to your level of understanding, whether you are a five year old child or a doctor of zoology. I can see the children of the future bouncing through this playground of knowledge, motivated by curiosity and the fun of instantly having their questions answered, being assisted by either a human or computer guide threading unique pathways through human knowledge. Imagine what it would be like, from birth, to be exposed to this type of media. A child would quickly become experienced in searching for and finding information on anything; relating ideas and weaving patterns of understanding, developing a form of thinking that would be highly conceptual. By today's standards, such a child would appear to be a genius. The impact on an entire generation could be dramatic. Our children's children may become the first genius generation.

Authoring books (or their electronic equivalent) on how to cope with the resulting generation gap should become a very profitable venture.

During this decade a new broadcast medium will also be born. Unlike today's TV networks, it will not depend on prime time viewing for its audience. The coupling of advanced recording techniques and microcomputer intelligence will allow us to scan the broadcast channels, 24 hours a day, seeking out subjects we find interesting, then recording them for viewing at our convenience. The broadcasters would be able to economically provide programming designed to satisfy both general and very specific viewer interests (narrow costing) - very similar to the book and magazine industry - since their potential viewing audiences are no longer time constrained. The number of broadcast channels may also increase dramatically in response to the individual's desire for specialized information.

This new "self selective media" may also become the source for everyday information such as supermarket prices, event schedules, daily newspapers, even electronic flea markets. From the home or the office we'd be able to quickly find almost any kind of information from a variety of sources to help us answer questions, make decisions and stimulate our curiosity and creativity.

Such free access to information will heighten concerns about privacy, however I feel most people will respond by becoming more open, specifically less concerned about what others know of us. A far greater challenge than privacy may be our individual capacity to meaningfully deal with fingertip availability of tremendous amounts of information. Without such a capacity, individuals may feel overwhelmed by everyday

demands to make reasoned choices. Severe indifference, withdrawal, compulsiveness, indecision and other symptoms might afflict large numbers of people who could no longer cope. Our ability to create new information technology that will allow people to easily organize information into pliable and meaningful patterns - an adaptive taxonomy of knowledge - may become as essential to cultural survival and growth as resolving the population, food and energy crisis we face today.

TO THE HEART OF THE ISSUE

At the core of the Information Age will be many-to-many communications. Individual computer capability tied into a sophisticated satellite, cable and broadcast based telecommunication systems which will permit millions of people to communicate simultaneously in vast interactive networks.

New activities and forms of governance will evolve based on many-to-many communication. Imagine playing Econogame. A simulated economy played interactively and currently by 50 million North Americans -- the 21st Century equivalent of Monopoly*. One would find themselves making better decisions in buying, sharing, selling, selecting jobs, by first simulating them through the game. The game would quickly assume all the attributes of the economy becoming a near perfect electronic image, anticipating needs and results from large numbers of individuals choices. In fact, it may become so hard to distinguish the "real" economy from its electronic image that people may cease making the distinctions. A new age of uncertainty begins unfolding that blurs what we have come to accept as physical reality and the dream like electronic images we create.

CLOSING REMARKS

As we move out of the industrialized era into the post industrialized information based economy, there will be tremendous adjustments in the labor force. I feel that today's inflation is only a sign of the dis-economies of transformation beginning to manifest itself in our society. As we move into the mid-80's we will go through some difficult economic times. Just as the world moved out of the agrarian age and into the industrial age, experiencing revolution and severe cultural disruption, so will the world undergo such change for the rest of this century.

Coming out of the century we will be a much different society than when we entered. It will be a society that is information rich, more diverse, more open and much more energy efficient. The advancements in microelectronics will have played a significant role in these changes just as the printing press helped transform medieval Europe. We will find ourselves linked to a culturally diverse global network - at the threshold of the Information Age.

Background of the Author

Alan Hald is a nationally known futurist and Chairman of the Board, of MicroAge® Computer Stores. His educational experience spans from the technological perspective of Rensselaer Polytechnic Institute where he studied philosophy, history and politics to the pragmatic managerial environment of Harvard Business School. His unique educational and professional background serves to make him a primary spokesman for the microcomputer industry, and the impact of this technology on our personal lives.

MANAGING INSTRUCTION WITH A MICRO - THE FIRST STEPS
Aly Gundlach - Coordinator of Computer Managed Instruction
Washington School District # 6
8610 North 19th Avenue
Phoenix, Arizona 85021

The microcomputer, recently thought of as little more than a toy, is rapidly coming into its own, as technology develops such modern wonders as chips and micro-processors. Whereas Computer Managed Instruction has been thought of as pure luxury for larger institutions and wealthy school districts, CMI is being considered more and more a necessity as educators are faced with pressures for accountability and increased paper work. Compared to astronomical costs of huge computers and programming, a micro, along with its software, can be purchased and installed for a cost similar to a good copying machine.

When watching a demonstration of Computer Managed Instruction, the process seems simple as pie, a panacea to our overwhelming problems of testing and evaluating on a continuous basis, grouping by instructional needs, and endless record keeping. However, as many of us are finding, the computer is unforgiving when our curriculum is not well organized, when the tests are less than satisfactory, when the system itself is not clearly defined as to the ultimate goals, procedures and expected results to be obtained from the acquired data. In other words, the microcomputer is capable of managing a very complex and sophisticated Instructional Program for many students, in a variety of programs, but it is necessary for the administration to do its homework before beginning to create the base files in the computer. Before discussing procedures for completing that "homework" it would be advantageous to describe how Computer Managed Instructors work in the form of PROJECT BASIC. Washington Elementary School District's system will be used as an example.

Washington utilizes the CUES (Continuous Uniform Assessment System) program as minimum competencies in three areas: math, reading, communication. CUES is made up of sequential objectives beginning in kindergarten and extending through the eighth grade. There are approximately 10-15 objectives per grade level in the Minimal Curriculum and an additional 8-10 objectives in the Extended Curriculum. For each identified objective, there is a mastery test usually comprised of 20 test items. This is called the CUES Mastery Test.

At each grade level, for each subject, there is a diagnostic test, or CRT, which assesses each minimal objective at that grade level. In the majority of cases each concept is evaluated by 5 test items or CRT. There also is a parallel CRT for Extended Objectives at each grade level.

The Computer Managed Instruction program, PROJECT BASIC, utilizes microcomputers to assist in test scoring, recording student progress, monitors student, grade and school progress, as well as assists in grouping children according to their instructional needs in math, reading, and written communication. Continuous assessment of student progress is included as part of our Instructional Model. Students requiring continued review, as well as those who have passed the Mastery Test, have their films automatically updated when their test results are entered in the computer. Instructional regrouping is part of the ongoing process and systematic review, as well as assessment, can be provided by computer printouts.

Current records for individual students, together with total grade levels and/or schools are reflected on the data files. Information on objectives still needing instruction and information on objectives instructed and assessed is available. Reports to parents will be run for all of the students in a school in math, reading and written communication.

The preliminary steps involve drawing a Model of Instruction, defining how the components of the curriculum flow into one another. If you are using a State or standardized test as an integral part of your curriculum, special attention needs to be given as to how that test modifies or influences instruction. Upon completion of the Model you must identify which areas will be directly reflected in the Computer Managed Instructional Program.

Most systems of instruction involve a diagnostic or screening device which allows identification of the specific subject matter to be presented. Following the teaching, a final assessment is administered to validate, to some degree, learning has taken place. Whether continuous assessment is integrated into the Instructional Model and whether there is systematic review and assessment depends on the degree of sophistication within the Model. The tests or assessments should, of course, be based on the specific subject matter to be instructed, the identified objectives, and the backbone of the curriculum.

In actuality, therefore, the Instructional Model is only as good as the curriculum. And the curriculum stands on the organization and the validity of the objectives.

You need to establish the subject areas you wish to manage in order of priority. Then you will need to designate your coding system. PROJECT BASIC requires a five digit number or letter to identify the objective. The first column represents the subject area, the second column the grade (usually the grade level at which mastery is expected), the third column represents the program, the fourth and fifth column represent the sequence of the objective on the file (usually within grade levels or programs).

The designation of program in the third column plays a key position in a system that will generate meaningful data. Often the decision to designate minimal, basic or State required objectives with one number or letter is elected. A second identifier represents remedial objectives, while a third might represent extended objectives. Still others might be used to identify Title objectives, Bilingual or Special Education objectives, where they differ from the minimal or basic skills identified originally. Remember that in the third column subject area is not represented since it is in the first column.

Coding Diagnostic Tests with a four digit representation follows a similar formula with the first letter or number being the subject area, the second column the grade level, the third column the program, and the fourth column the sequence.

The Mastery tests are filed by objective so they carry the same number as does the objective. Prior to entering your objective on the file, you will have to determine whether or not your objective tests are teacher judgment or multiple choice. If they are multiple choice, the correct responses will need to be determined.

The coding of teachers, classes, and/or disks have less option and can be done expediently once the decision of how your students should be organized on disks is made. There are essentially two options - one to organize by individual class (i.e. Mrs. Jones' Reading, Mr. Brown's Reading) and second is by grade level. There are advantages and disadvantages to both. The major advantage to organizing by class is that data is only for the small specific group of students. The advantage of grouping by grade level is that the data allows for flexible grouping by instructional needs and, secondly, that the data files are ongoing and do not need to be created each year to reflect class rosters.

Perhaps the most critical coding considerations you will encounter is that of the student number. PROJECT BASIC requires a six digit code. The first two digits identify the school, the third digit represents the program.

It is not unlikely that as many as twenty or twenty-five programs may be monitored; particularly, Special Education is broken down into all the sub-categories concerning LD Resource, LD Extended Resource, LD Self-Contained (see appendix for example of Program Code). The fourth digit should represent

either year of entry or grade level. Since the grade level of the student changes each year, this number is different for each successive grade. For example, in the year the computer is implemented. Billy is in grade 3, so his fourth digit is three. In the second year of implementation, three is "coded" to mean grade four, two is "coded" to mean grade three, and so forth. A memo to all users and operators at the beginning of each year with grades and corresponding "coded" numerals simplifies this.

PROJECT BASIC is presently limited to 50 students per disk. For this reason, each student on each disk must have a different last two digit sequence number in the fifth and sixth columns. All individual reports, therefore, can be obtained by using the final two digits.

Regardless of the software you might use, it is absolutely essential that sufficient time and thought go into developing a coding system that represents your instructional program. Because of the inherent generic nature of PROJECT BASIC you will find many options open to you. If you are attempting to convert an already published curriculum with a coded retrieval system to PROJECT BASIC you will often find that by preceding the title with the original number, the transition for staff is easier. It will be necessary to develop some kind of administrative time line. In the preliminary stages of developing a Computer Managed Instructional program, it is so important that an indepth evaluation of the curriculum be made. Such consideration as to how are the tests to be written, answered, scored and used as criteria

for setting up instruction must be made. For example, an initial diagnostic instrument can assess all or a portion of objectives at a grade level. It might also span several grade levels. Perhaps, as an alternative, it will evaluate students in specific skill areas, crossing grade levels. Addition, fractions, telling time, sequencing of events, spelling application, are all examples of skill areas. Choices for computer scored tests are limited to multiple choice or true and false. All other alternatives, such as short answer, essay, oral, etc. must be entered on the file as teacher judgment. Although these responses can be "coded" for computer scoring. This process is time consuming and complicated, but very useful when the data obtained is necessary.

When the Diagnostic Tests are created on the PROJECT BASIC files there is an option of entering an assignment for each item on the test. Therefore, when the test is computer-scored, the assignment is printed out for each item missed. If this is to be done initially, then ample time to match objectives and/or test them with assignment needs to be allowed for. In substantial numbers of districts, this information is added to the files at a later date, so that implementation of CMI can occur earlier. The first and most critical task on your timeline then is organization and critical evaluation of your curriculum. The following tasks should be considered within your timeline.

On the following page is an Administrative Checklist provided in the PROJECT BASIC DOCUMENTATION by Evans-Newton Corporation. Each item should be carefully considered to be sure that it

reflects your Instructional Model, it is not in conflict with another item, and perhaps even more important, that it will allow for growth and development as your curriculum and administration needs expand or change.

RECOMMENDED TIMELINE CONSIDERATION FOR CMI

1. Organization & evaluation of curriculum
2. Coding your instructional system
3. Identification of subject area, grade levels, schools, etc. in order of priority to be managed by computer
4. Approximate dates of equipment arrival
5. Administrative and principal inservice
6. Operator inservice and training time
7. When and how many class or grade disks are to be completed
8. Teacher and staff inservice
9. When should all Diagnostic Tests be completed and base student files completed
10. Procedures and schedule for Objective Test survey
11. When should cross referencing of instructional materials for assignments to be correlated to Diagnostic Test items be completed if not done in 1 and 2 above
12. When should grade, school and district status reports be generated

Although only listed on the Timeline Consideration, it is absolutely imperative to the success of CMI that great emphasis be placed on Teacher Training and Inservice, Staff needs to understand the Instructional Model, separate from the computer.

Then they need to be trained in how the computer will provide them with time-saving, accurate reports, based on continuously updated files. They, along with the administration, must also be trained in how to use the reports, what the data means. Parents, too, need to be brought into awareness and involvement. And last, but certainly not least, the students need to be inserviced on the hows (to use the electronic pencils, cards) and whys (keep track of what still needs to be learned & reviewed). An ongoing districtwide CMI Inservice Program needs to be developed in conjunction with CMI itself.

DEVELOPING INSERVICE FOR COMPUTER MANAGED INSTRUCTION:

The Second Step

The need for inservice training within the school district cannot be stressed enough. You can have the world's best curriculum -- materials, testing instruments, computer programs, etc. -- but if there is not a districtwide common understanding you will end up in utter chaos or with everyone having a different understanding of what is going on.

Initially, the district needs to consider the position of a full time person responsible for computer managed instruction with enough authority to deal with key persons in curriculum, program implementation/staff development and principal supervision.

The school district needs to provide training for operators under the direction of the organization providing the software. At the same time, intensive work with administrators needs to begin in preparation for a well organized computer managed instruction program. This training should specifically focus on the curriculum itself, with emphasis on those components that will be computer managed; for example, diagnosis, instructional grouping, and mastery testing.

These questions will need to be addressed prior to implementation:

1. Who will be responsible for inservice training at various levels? (Administration - Principals - Teachers - Students)
2. Are the materials field tested, distributed, and are

Developing Inservice for Computer Managed Instruction

the teachers familiar with the administration of them?

3. Are the principals aware of what information is available from computer printouts?
4. Are the operators fully trained?
5. Are parents aware that CMI is being implemented?
6. Are the students aware that the computer is being used and are they trained to take tests that are computer scored?

MICROCOMPUTERS IN EDUCATION

Lorraine Mecca
2612 Croddy Way
Santa Ana CA 92704

C P U

Microcomputers contain one primary integrated circuit (IC) called a microprocessor. The microprocessor directs the flow of data. The piece of equipment that contains the microprocessor chip is called the CPU. CPU is short for Central Processing Unit. Microcomputers are classified by their CPU's. There are two very popular microprocessors that can be found in most personal computers: the 280* chip and the 6502. The 280 has some programming advantages over the 6502, but the 6502 is becoming more popular with new personal computer manufacturers. Each of these CPU's can process eight BITS of information at a time.

A recent development in personal computers is the 16 BIT processor. This CPU is capable of handling twice as much data at one time. However, there are very few software programs available today to take advantage of the increased capacity of the 16 BIT processor.

BITS AND BYTES

If you have ever studied computer sciences or if you have an engineer as a close friend, you know that computers can only distinguish between two signals. A computer sees all data as either a one or a zero. Each one or zero is called a BIT. If you string eight BITS of information together you have a BYTE. A BYTE is about the smallest unit of information with which most people are ever concerned. An example of a BYTE would be the letter F or the number 4. It takes eight BITS of information, eight ones or zeros, to create a CHARACTER. A CHARACTER can be any number, letter, or symbol. The CPU's found in most popular microcomputers process information eight BITS or one BYTE of data at a time. If the microprocessor in a personal computer can process more than eight BITS at one time, then a BYTE for that particular CPU will be larger than a BYTE in an eight BIT processor.

TERMINAL

There is nothing deadly about a computer TERMINAL. It is simply the end of the system. The TERMINAL is the vehicle for interaction between the CPU and the individual using the system. A TERMINAL has two parts, the keyboard and the screen. The screen is often called the VIDEO MONITOR or the CRT (Cathode Ray Tube). Some personal computers are designed to use a standard television screen to display data, while others require a specially designed VIDEO MONITOR.

The keyboard of a terminal is very similar to that of an electric typewriter. There are a few extra keys on the keyboard and there is often a numeric key pad like that of a calculator. When you want to enter data into the computer you just type the information you want the CPU to process. This is called DATA INPUT because you PUT DATA INTO the computer.

Lorraine Mecca
2612 Croddy Way
Santa Ana, CA 92704

**EVERYTHING YOU EVER WANTED TO KNOW ABOUT PERSONAL COMPUTERS
BUT DIDN'T KNOW HOW TO ASK**

WHAT IS A PERSONAL COMPUTER?

A personal computer is a tool. You should think of a personal computer the way you would think of a typewriter or a lawnmower. Tools are used to save time and energy, and that's what a computer can do. If you have enough time and energy to do everything you want to do, you don't need a computer. But if there isn't enough of you to go around, then a personal computer may be exactly what you need.

The most advantageous thing about a tool like a personal computer is that it can be used in so many diverse applications. The same computer system one individual uses to turn on sprinklers or monitor family energy consumption can be used by someone else to do sales analysis or write poetry.

WHAT DOES A COMPUTER DO?

A computer processes data. "Fine," you say. "The next time I need my data processed I'll let you know." Try not to be alienated by computer terminology. Data is just another way of saying information and processing is another way of expressing movement. If a large percentage of your time is spent doing repetitive but necessary tasks then a personal computer could provide you with organization and speed.

A functioning computer system is part **HARDWARE** and part **SOFTWARE**. The **HARDWARE** part of a computer system is the tangible electronic equipment. The **SOFTWARE** part of a system is a series of electronic instructions that directs a computer to perform a specific task.

Today's personal computer systems (there were none available before 1970) have a few basic pieces of **HARDWARE**. They are the CPU (Central Processing Unit), the Terminal, the Disk Drive, and the Printer.

SOFTWARE is to a computer system what music is to a stereo system. There is no reason to have all that state-of-the-art gear if there aren't any **SOFTWARE** programs you want to use. **SOFTWARE** can be divided into three basic categories, though there can be some overlapping. They are Utility Programs, Operating Systems and Application Software.

DATA OUTPUT is when the computer takes information that has been processed by the CPU and moves the data out of the computer. The output data can be seen on the screen of the terminal. Data can be output to a printer, or it can be saved for later use. There are several other pieces of **HARDWARE** that can be used for **DATA OUTPUT**.

DISK DRIVE

A **DISK DRIVE** is an input/output device, just like a terminal. A **DISK DRIVE** is used for storage of programs, information or files. Data can be output to the **DISK DRIVE** to be saved for later use. Files and programs that have been saved on the **DISK DRIVE** can be input to the CPU for processing.

A **DISK DRIVE** can be compared to a tape recorder in a stereo system. In fact, some early personal computers used only cassette recorders for data storage. A **DISK DRIVE** has a highly sensitive head that reads and writes information. There are three kinds of **DISK DRIVE**.

A **HARD DISK DRIVE** is large and expensive and is seldom used in home computers. Hard Drives can hold many Megabytes of data. (A Megabyte is 1,000,000 characters.) Data is stored on a magnetic medium housed in a hard protective case. Hard Drives are necessary for most small business applications. Hard Disk Drives come in various sizes and shapes.

A **FLOPPY DISK DRIVE** gets its name from the 8" diskette that is used for data storage. Unlike the hard disk that is used in a Hard Drive, a Floppy Disk Drive uses a flexible magnetic disk that is always handled by a protective paper covering. Since the diskette is flexible, it is called a **FLOPPY**.

A **MINI FLOPPY DRIVE** works like a **FLOPPY DISK DRIVE**, but it is smaller. A **MINI FLOPPY** uses 5 1/4" diskettes. Since the diskettes are smaller, the amount of information that can be stored is also smaller.

Storage space for a personal computer can be doubled by adding an additional disk drive to your computer system. Storage can also be increased by changing density. In a single density system information is written on the disk at certain intervals. These intervals are determined by the integrated circuit board that controls the disk drives. Single density disk drives are very reliable but are limited to about 250,000 bytes of information on an 8" drive and 100,000 on a mini-floppy 5 1/4" drive.

If a dual density, or quad density controller card replaces the single density printed circuit board that controls the disk drive, then storage is dramatically increased. This increased density requires greater precision in reading and writing information. It also requires a higher grade of diskette than is required by a single density drive. This technology is not always available for all personal computers, nor is it required for most applications.

PRINTERS

A **PRINTER** is a peripheral device that takes electronic data and transforms it into printed symbols. (A device that attaches to a computer is called a **PERIPHERAL**) Printed information is called **HARD COPY**.

There are three types of **PRINTERS**: Thermal, Impact, and Ink Jet. The prices of **PRINTERS** are proportional to their speed and print quality and options. Some printers have keyboards and can also function as a terminal or typewriter when not in a print mode.

Thermal or electrostatic printers form images by heat transfer to treated paper. They are usually inexpensive but somewhat limited. These printers must use only treated paper. The paper is often narrower than a standard 8 1/2 X 11" page.

An Impact Printer forms a letter on a page by making an inked impression. There are two classifications of Impact Printers: Dot Matrix and Letter Quality.

Dot Matrix printers are those that are normally associated with computer generated **HARD COPY**. The printed characters are made up of many tiny dots. Dot Matrix printers vary in price with the number of characters printed per second and the number of dots per character. Dot Matrix printers are fast and reliable, but have that tell tale computer generated look. They are perfect for most personal and small business applications.

A Letter Quality printer is every typist's dream come true. These printers are also called Daisy Wheel printers. Printing occurs when a single character impacts the page to form the image. These printers are often slower than dot matrix printers but faster than any typist ever clocked. A Daisy Wheel printer can be very expensive. They are used whenever quality is demanded.

Ink Jets printers actually spray ink on a page. They are extremely fast and very expensive.

SOFTWARE

Personal computer software can generally be divided in three categories: **UTILITY PROGRAMS**, **OPERATING SYSTEMS**, and **APPLICATION SOFTWARE**. These divisions are not always absolute.

For the most part **UTILITY** programs are tools used by programmers to write or modify software. Some examples of utility programs are:

Assembler	Compiler
De-Bugger	Editor
Line Renumberer	Merge
Sort	

The higher level languages require few or no utility programs programs as they are included within the language.

OPERATING SYSTEMS enable the CPU to communicate with other parts of the computer system and the outside world. The operating system tells the processor where to look for incoming signals and where to display that data. They enable the disk storage system to function and the printer to produce hard copy. The key work to understanding the function of an operating system is **INTERFACE**. The operating system allows the microprocessor to interface with the computer peripherals and human beings.

Some examples of operating systems are:

CP/M *	MP/M *
DOS	MicroSoft BASIC *

APPLICATION SOFTWARE can be any program that directs the computer to perform a specific task. There are programs that drill students in multiplication and division, and others that calculate payroll. Computer games are application software as are programs to compose music or develop student courseware. Application software is the most plentiful category of software.

Application software is written for a particular CPU and has minimum memory requirements. Most application software will list the required system configuration on the outside of the package. The cost of application software varies with the sophistication of the program.

- * CP/M and MP/M are trademarks of Digital Research
- * MicroSoft is a trademark of MicroSoft

SOFTWARE - TRS-80 I, III

TRS-80 educational software developed by Tandy is available at Radio Shack stores. It is the practice of most Radio Shack stores to sell only their brand name products. Information about educational software by other vendors is not available through Radio Shack despite the fact that TRS-80 software is more plentiful than software for other personal computers.

Sources for TRS-80 software can be found in computer magazines and trade journals, especially those specializing in this unit. Educational programs can also be purchased from the following organizations.

Computer Programming Unlimited
6712 Langston Drive
Austin, TX 78723
(512) 928-2626
Composition

Micro Learningware
Box 2134
N. Monkato, MN 56001
(507) 625-2205
General Education

Educational Programs
P.O. Box 2345
West Lafayette, IN 47906
(317) 463-4778
Math

Minnesota Software, Inc.
5422 Fisher Street
White Bear Lake, MN 55110
(612) 426-0916
General Education

Educational programs available through Radio Shack are listed on the following page.

SOFTWARE - BELL & HOWELL/APPLE II

Educational software for the Bell & Howell and Apple II computer is available from the following vendors and can often be found at local retail computer stores.

Apple Computer, Inc.
10260 Bandley Drive
Cupertino, CA 94017
(408) 885-1010
Representative Educational Programs

Bell & Howell
7100 McCormick Rd.
Chicago, IL 60645
(312) 673-3300
GENIS I, MARK PILOT

COMPRESS
P.O. Box 102
Westworth, NH 03282
(603) 764-5831
Population, Genetics, Statistics

Computer Solutions
5135 Fredericksburg Rd.
San Antonio, TX 78229
(512) 341-8851
Instruction in BASIC

Cook's Computer Company
1905 Bailey Drive
Marshalltown, IA 50158
(515) 752-8279
Elementary Math

CONDUIT, University of Iowa
100 LCM
Iowa City, IA 52242
(319) 353-5789
Assorted College Level Subjects

Creative Discount Software
256 South Robertson, Suite 2156
Beverly Hills, CA 90211
(800) 824-7919
Elementary Math

EduTek
P.O. Box 11354
Palo Alto, CA 94306
(415) 325-9965
Elementary Math

EduWare
22222 Sherman Way #102
Canoga Park, CA 91303
(213) 346-6783
Math, Spelling, Reading, Statistics

Instant Software
Petersborough, NH 03458
(800) 258-5473
Elementary Math Games

Instructional Micro Systems
1249 Greentree Lane
Narberth, PA 19072
(215) 664-1207
Reading - all levels

Millsiken Publishing
1100 Research Blvd.
St. Louis, MO 63132
(314) 991-4220
Elementary Math, Medical Ed.

Minnesota Education Computing Consort.
2520 Broadway Drive
Lauderdale, MN 55113
(612) 376-1117
General Education

Micro Users Software Exchange
7112 Darlington Drive
Baltimore, MD 21234
(301) 661-8513
Nuclear Simulation

SOFTWARE - BELL & HOWELL/APPLE II

POWERSOFT
P.O. Box 157
Pitman, NJ 08071
(609) 589-5500
Engineering, Math, Statistics

Programs for Learning, Inc.
P.O. Box 954
New Milford, CT 06776
Chemistry

Rainbow Computing Inc.
9719 Resida Blvd.
Northridge, CA 91324
(213) 349-5560
General Education

Software Factory
515 Park Street
Anoka, MN 55303

General Education

TYC
40 Stuyvesant Manor
Geneseo, NY 14554
Mind-memory Improvement

Programma International, Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 954-0240
General Education

Program Design, Inc.
11 Idar Court
Greenwich, CT 06830
Reading, Writing, Spelling

Softagon
P.O. Box 744 M
Morristown, NJ 07960
(201) 539-3770
Elementary Reading

Sterling Swift Publishing Co.
P.O. Box 188
Manchaca, TX 78652
(512) 444-7570
Literacy Materials

SOFTWARE - COMMODORE PET

A listing of available software for the Commodore PET computer is currently being compiled by Commodore and will be released in approximately two months. Commodore has recently appointed an Education Marketing Coordinator, Dorothy Engel. Ms. Engel is compiling this listing and is interested in hearing from any individual or school who has developed educational application software. Please direct responses to:

Commodore
950 Rittenhouse Road
Norristown, PA 19403
Attn: Dorothy Engel, Ed. Marketing Coordinator
(215) 666-7950

PET software is currently available from the following sources:

Softswap
San Mateo Office of Education
333 Main Street
Redwood City, CA 94063
(415) 364-5600

Public domain software is supplied free of charge, if you supply the diskettes.

A Title 4C program led to the development of approximately 75 programs for use in education. For program listings contact:

Microcomputers in Education
Robbinsdale Area Schools
Independent School District 281
4148 Winnetka Ave. North
Minneapolis, MN 55427
Dick Maus or Bill Hick
(612) 533-2781

Some other sources of PET educational software are:

Suny Storybrook
New York, NY
Dr. Ludwig Brown
(516) 246-8418

Regency Education System
Dallas, TX
Tom Shephard
(214) 233-7304

Micro Ed
Minneapolis, MN
Tory Esbensen
(612) 926-2292

R.W. London Computer Central
Minneapolis, MN
Betsy Tronsdal
(612) 830-1122

COMPARISON CHART

MANUFACTURER	TANDY	APPLE/BELL&HOWELL	COMMODORE	ATARI	
MODEL	TRS-80 III	II	PET	400	800
PRICE	\$669.00	\$1330.00	\$995.00	\$500.00	\$1080.00
ENCLOSURE					
TABLE TOP DESK	YES OPT. \$56,179	YES OPT \$149	YES OPT \$149	YES NONE	YES NONE
CPU					
TYPE	Z80*	6502	6502	6502	6502
BITS	8	8	8	8	8
BUS	EACH MANUFACTURER PROVIDES DIFFERENT METHOD OF EXPANSION				
RAM					
STANDARD	4K	16K	16K	8K	16K
MAXIMUM	48K	48K	32K	16K	48K
STANDARD PERIPHERALS					
DISK DRIVE	OPT	OPT	OPT	NONE	OPT
TERMINAL	STD	OPT	STD	OPT	OPT
GRAPHICS	YES	YES	YES	YES	YES
SCREEN CONFIG.	64 X 16	40 X 24	40 X 25	40 X 20	40 X 20
LOWER CASE	STD	OPT	STD	STD	STD
PRINTER	OPT	OPT	OPT	NONE	OPT
STORAGE					
CASSETTE	STD	OPT	OPT	OPT	STD
FLOPPY					
5 1/4"	OPT	OPT	OPT	NONE	OPT
BYTE CAPACITY	134,000	130,000	180,000	NONE	90,000
OPERATING SYS.	TRS-DOS	DOS 3.3	BASIC	BASIC	DOS
PROGRAMMING LANGUAGES					
BASIC	STD	STD	STD	STD	STD
FORTTRAN	OPT	OPT	OPT	NONE	NONE
PASCAL	OPT	OPT	OPT	NONE	END 81

COMPARISON CHART

MANUFACTURER	TANDY	APPLE/BELL&HOWELL	COMMODORE	ATARI	
MODEL	TRS-80 111	11	PET	400	800
PRICE	\$669.00	\$1330.00	\$995.00	\$500.00	\$1080.00
ENCLOSURE					
TABLE TOP	YES	YES	YES	YES	YES
DESK	OPT.\$56,179	OPT \$149	OPT \$149	NONE	NONE
CPU					
TYPE	280*	6502	6502	6502	6502
BITS	8	8	8	8	8
BUS	EACH MANUFACTURER PROVIDES DIFFERENT METHOD OF EXPANSION				
RAM					
STANDARD	4K	16K	16K	8K	16K
MAXIMUM	48K	48K	32K	16K	48K
STANDARD PERIPHERALS					
DISK DRIVE	OPT	OPT	OPT	NONE	OPT
TERMINAL	STD	OPT	STD	OPT	OPT
GRAPHICS	YES	YES	YES	YES	YES
SCREEN CONFIG.	64 X 16	40 X 24	40 X 25	40 X 20	40 X 20
LOWER CASE	STD	OPT	STD	STD	STD
PRINTER	OPT	OPT	OPT	NONE	OPT
STORAGE					
CASSETTE	STD	OPT	OPT	OPT	STD
FLOPPY					
5 1/4"	OPT	OPT	OPT	NONE	OPT
BYTE CAPACITY	134,000	130,000	180,000	NONE	90,000
OPERATING SYS.	TRS-DOS	DOS 3.3	BASIC	BASIC	DOS
PROGRAMMING LANGUAGES					
BASIC	STD	STD	STD	STD	STD
FORTTRAN	OPT	OPT	OPT	NONE	NONE
PASCAL	OPT	OPT	OPT	NONE	END 81

MICROCOMPUTERS IN EDUCATION-FUNDING

Henry Dahlberg, Ed.D.
Director of Special Projects
Prescott Unified School District #1
Prescott, AZ 86301

"Possible local, state, and federal sources of funding for computer programs will be discussed, as well as the modus operandi for securing those funds."

I am sure that we would all love to attend a funding session sometime where the presenter was seated at the front of the room passing out checks in the amount that you specified. Unfortunately this will not be that type of session!

I am also sure that many of you here have a favorite source, but few would have the temerity to share it with the competition. I do not come concealing any such source, but will attempt to enumerate several potential sources, many of which we have found helpful in the funding of our Computer Literacy Project in Prescott. The Project in Prescott will have received funding from numerous sources, equaling approximately one quarter of a million dollars, by the end of the third Project year. This funding has been garnered from both Public and Private sources.

It is essential, when you are starting any new project, to define for yourself the meaning of success. If for you, success is a single computer in your classroom, then that goal is fairly easily achieved. However, if you are looking at a broad program, with extensive and significant student impact, then that requires considerably more effort and time. The key is to start. The hardest part of beginning is to begin. Start small, but start!

Remember that any new project is a marketing job, you must sell Computer Education the same way that any entrepreneur markets his goods. Work closely with administrators, with the School Board, with teachers, and parents. If someone says no to a request...Simply assume they have not understood your question. Be persistent. There is money. What is essential is to achieve the "correct" ordering of priorities.

At some point in time, in any attempt to secure funds, something must be written down. When writing your proposal: know your subject, point out what you have already done, tie your project into basic skills. Finally, and most importantly, start early enough so that you can submit a draft to that potential source of funding, and then listen to what they say, and make the necessary changes. Don't let pride in your own prose get in the way of spelling out what the funding agency wishes to read.

Some sources of funding which have been successfully used by the Prescott Project or other Projects with which I am familiar, are as follows:

1. Title I of the Elementary and Secondary Education Act provides for compensatory education. The funds are designed to supplement local funding providing a variety of programs and service. Title I provides for equipment purchases, resource center materials, tutorial aide assistance: all areas of great interest to anyone starting a computer project.
2. ESEA Title IV-B provides for the purchase of instructional materials and equipment for use in teaching academic subjects. Both Title I and Title IV-B are a source for purchase of non-consumable materials and capital outlay items.

3. ESEA Title IV-C is a third source of funding generally more difficult to secure because it is a highly competitive funding source. Writing a competitive Title IV-C proposal requires many months. However, there is a provision in Title IV-C for the Adoption of validated computer (and other) projects. I am happy to say that there is a Computer Literacy program entitled The Gifted Child Project, in Tallahassee, Florida, which was validated just this past December 23. To apply for Adopter funding is a rather simple task. The proposal itself requires no more than a day or two to write, and the \$5,000 maximum would supply some equipment, and just as important, training for staff. There are a few other validated computer projects in areas of math instruction and classroom management.
4. The Vocational Education Act is another source for capital outlay funds, particularly for computer programs in the business teaching area.
5. Gifted Program funds, from either the State or local District are an additional source which can provide some hardware, but mainly software.
6. Parent Teacher Associations have probably purchased as many computers as any other single source in this country and will continue to be an important source of funds as the need for computers becomes more widely recognized, and as the cost of computing systems continues to decline.
7. Other sources would be rummage sales (Proceeds from such sales have purchased two computers in the Prescott District.), local industry and business and interested parents.
8. The final and most important source is your own local District. This is a source which you cannot fail to cultivate simply because you have "hot leads" elsewhere. Ultimately, these computers will become the District's responsibility: repair and maintenance of equipment, training of staff, recognition of the importance of computers in curriculum, payment of salaries for teachers for whom computer instruction may become a full-time responsibility. All of these needs must be met at the local level.
9. While of no direct monetary benefit, it is crucial that the District's local patrons be made aware of the Computer Literacy or other Computer Programs that are occurring in the school system. One technique which we have used to our advantage in the Prescott Schools is to offer a no credit, no cost, no grades course, extending over three nights for interested parents and patrons. This course has been highly successful. It is taught at no cost to the District by volunteer teachers, and begins to create that reservoir of knowledgeable patrons which might spell

the difference between nurturing or abandoning the computer project as district budgets become increasingly tight.

In summary, there is money, if a District recognizes the importance of computers in the instructional process. This recognition may be slow in coming, but it depends to a large extent on the ability of people, such as you, to successfully market what Judith Edwards referred to in her Luncheon Talk as the "Vegematic of the 1980's!"

TEMPE DISTRICT'S ELEMENTARY COMPUTER PILOT PROJECT AT MEYER SCHOOL

Polly Mack
Teacher
Tempe School District #3
Tempe, Arizona 85282

Marilyn J. Latham
Pace Teacher
(Gifted Program)
Tempe District #3
Tempe, Arizona 85282

Milton Zuroff
Principal
Meyer School
2615 S. Dorsey Lane
Tempe, Arizona

In the spring of 1980, members of the Meyer P.T.O. Executive Board developed a proposal for the Tempe Board of Education suggesting the need for a pilot computer program, which would involve elementary students in Computer Assisted Instruction (C.A.I.) and computer literacy. In other words, in addition to the computer being used in the regular classroom to assist teachers in teaching basic skill areas, the computers would also be used to teach students problem solving and computer awareness including the uses of computers in our society, its capabilities and limitations, its history, and its possible uses in the future.

The proposal was approved by the Tempe Board of Education, on May 7, 1980, and the project began with the purchasing of the following equipment, which was housed in a special room at Meyer School called, "The Computer Center:"

- * 4 TRS-80 Level II Microcomputers with one line printer, expansion interface, and network.
- * 4 Texas Instruments 99/4 home computers, and speech synthesizer.

As we have stated previously, part of the purpose of the computer center was to provide C.A.I. , and educational software was purchased to accompany each system in different skill areas within the instructional program.

Beginning in September, 1980, students in grades 1-6th at Meyer

Elementary School, were either scheduled into the computer center for various times during the week, or the computers were taken directly into the classrooms by the teachers. The instructional program was supplemented through software designed to reinforce basic skill areas such as math, reading, language arts, and problem solving. Student progress was (and continues to be) monitored by the computer through Computer Managed Instruction (C.M.I.) and through parent assistance. (Parental assistance is needed to record the computer generated scores and average response times).

In addition, the following programming skills were presented to students in grades 4-5-6:

*Flow charting

*BASIC commands including:

PRINT	END
GOTO	CLS
LIST	CALL CLEAR
RUN	INSERT
EDIT	DELETE
CALL SOUND	RND
CALL SCREEN	RANDOMIZE
CALL HCHAR	FOR/NEXT
CALL CHAR	
CALL UCHAR	
CALL SAY	

The Meyer Computer Center has provided the following services for parents, students, teachers, and administrators within the Tempe Elementary School District and to other neighboring districts:

- Inservice Education
- Career Education for students
- Computer Literacy and Programming for elementary school children
- Evaluation of software designed to reinforce basic skill areas

In addition, the Meyer School Computer Center can be considered a model classroom and is open for visitation on two half days per week: Thursday afternoon's, and Friday mornings. For further information, please contact:

Marilyn Latham
Project Coordinator

COMPUTER RESOURCE CENTER RATIONALE AND PROCEDURES

THE FUNDAMENTAL RATIONALE OF THE COMPUTER ASSISTED INSTRUCTION (CAI) PROGRAM IS TO INDIVIDUALIZE THE LEARNING ACTIVITIES IN PRESENTED SUBJECT MATTER AND TO FACILITATE INDIVIDUALIZATION OF LEARNING FOR EACH STUDENT. LIKE BOOKS, COMPUTERS SHOULD SERVE BOTH THE STUDENT AND THE TEACHER AS ADDITIONAL AIDS FOR LEARNING AND TEACHING. THEY SHOULD TAKE SOME OF THE ROUTINE LOAD FROM THE TEACHERS AND GIVE THEM AN OPPORTUNITY TO SPEND MORE TIME WITH INDIVIDUAL STUDENTS, AND ALLOW THE STUDENT TO WORK ON HIS/HER OWN PARTICULAR LEARNING PROBLEMS. CAI CAN BE HIGHLY MOTIVATING EXPERIENCE FOR MOST STUDENTS AND IN MANY CASES, HELP THEM ACHIEVE LARGE GAINS IN THEIR PROGRAM SCORES. A WELL DESIGNED CAI PROGRAM WILL PREPARE, ADMINISTER, CORRECT AND REPORT ON THE SUCCESS OF AN INDIVIDUALIZED LESSON FOR EACH STUDENT. THE CLASSROOM OR RESOURCE TEACHER, THROUGH THE USE OF THE COMPUTER REPORTS CAN PINPOINT THE AREA OR AREAS WHERE THE STUDENT IS HAVING DIFFICULTY AND PROVIDE PRESCRIPTIVE REINFORCEMENT MATERIALS IN THE CLASSROOM INSTRUCTION DESIGNED TO IMPROVE THE STUDENT'S PROGRESS.

IT IS IMPOSSIBLE TO PRESENT A TOTAL PICTURE OF CAI AND THE VARIOUS MANAGEMENT SYSTEMS POSSIBLE WITHIN THE FORTY OR FIFTY MINUTES ALLOCATED FOR THIS TYPE PRESENTATION. IN ORDER TO SPEED UP THE MATERIAL FLOW, AN OUTLINE FORM OF THE PRESENTATION HAS BEEN PROVIDED, WITH CHARTS AND MODELS TO AID IN A RAPID UNDERSTANDING OF CRITERIA AND NEEDS IN BUILDING A MANAGEMENT STRUCTURE FOR A COMPUTERIZED RESOURCE CENTER. IT IS EXPECTED THAT THE PARTICIPANTS WILL GAIN AN INSIGHT INTO THE PROBLEMS, PERSONNEL, AND INTEGRATION OF CURRICULUM WITH CLASSROOM ACTIVITIES TO ACHIEVE MAXIMAL RESULTS WITH MINIMAL EFFORTS AND EQUIPMENT. FIRST AND FOREMOST, THE IDEA OF INDIVIDUALIZED INSTRUCTION MUST BE EXPLORED IN CONJUNCTION WITH THE LEARNING OR INSTRUCTIONAL CYCLE.

PRESCRIPTION, DRILL, PRACTICE, AND CHECK

MANAGEMENT SYSTEMS FOR COMPUTER ASSISTED INSTRUCTION

PRIMARY INSTRUCTIONAL USES

1. LEARNING ABOUT COMPUTERS
 - A. COMPUTER SCIENCE
 - B. DATA PROCESSING
 - C. COMPUTER TECHNOLOGY
2. LEARNING WITH COMPUTER SUPPORT
 - A. COMPUTER ASSISTED INSTRUCTION
 - B. COMPUTER AIDED INSTRUCTION
3. ADMINISTRATION OF LEARNING
 - A. COMPUTER MANAGED INSTRUCTION
 - B. COMPUTER MANAGED TESTING AND EVALUATION
 - C. COMPUTERIZED RECORD KEEPING

BASIC INSTRUCTIONAL FUNCTIONS

1. PRESENTATION OF MATERIAL
2. REVIEW OF MATERIAL AND REMEDIATION
3. PROVIDE ENRICHMENT SITUATIONS

INSTRUCTIONAL TECHNIQUES

1. SIMPLE TUTORIAL
2. DRILL AND PRACTICE
3. DIALOGUE
4. PROBLEM SOLVING
5. GAMES
6. SIMULATION EXERCISES
7. INQUIRY SITUATIONS

EXPECTATIONS

1. INCREASE TEACHER PRODUCTIVITY
2. INCREASE STUDENT ACHIEVEMENT
3. INDIVIDUALIZE INSTRUCTION
4. IMPROVE RETENTION AND ACQUISITION
5. REDUCE PERSONNEL NEEDS

PROBLEMS IN ACCEPTANCE OF COMPUTERS FOR THE CLASSROOM

1. MISINFORMATION
2. TEACHER REPLACEMENT THEORY
3. COST-INITIAL AND CONTINUAL
4. LACK OF ADMINISTRATIVE KNOWLEDGE
5. COMPUTER LITERACY
6. BRAIN BANKS VS RE-INVENT THE WHEEL CONCEPT
7. MISUSE OF MANAGEMENT SYSTEMS OR ABSENCE OF CONTROL SITUATION

PURPOSE FOR COMPUTERS IN THE CLASSROOM

1. TO IMPROVE LEARNING SITUATION
2. RELIEVE TEACHER ROUTINE FUNCTIONS
3. CREATE STUDENT INTEREST
4. INDIVIDUALIZE INSTRUCTION
5. LESSEN PEER PRESSURE
6. PROVIDE A NON-HOSTILE ATMOSPHERE
7. PROVIDE AND KEEP ACCURATE RECORDS

IMPLEMENTATION STRATEGIES AND METHODOLOGY

1. CLASSROOM USE
2. RESOURCE CENTERS
 - A. PULL OUT SYSTEM
 - B. TOTAL CLASS PARTICIPATION
 - C. TUTORIAL AND SPECIAL SITUATION
 - D. OTHER
3. PARENT AND ADULT USE

MICRO AND MINICOMPUTER MANAGEMENT SYSTEMS IN THE RESOURCE CENTER CONSIDERATIONS

1. INVESTIGATE AND FIND OUT DESIRED RESULTS FIRST
2. CONSIDER A SYSTEM FLOWCHART AND NEEDS SURVEY
3. COST AND FUNDING SOURCES FOR IMPLEMENTATION
4. EXPANSION AND ADDITIONS TO SYSTEM PLANS
5. AVAILABILITY OF CURRICULUM AND SOFTWARE NEEDS
6. TEACHER INSERVICE AND PERSONNEL NEEDS

CAI Skills Area and Classroom Instruction

The resource teacher and the participating classroom teacher were responsible for the instruction of the total class in the CAI lab. The original CAI concept of having just youngsters reporting to the lab for drill and practice was modified by the management system to require that the students be accompanied to the lab by their teacher, and he/she would use the skills area of the lab to instruct youngsters with the help of the resource teacher.

This procedure gives an immediate eight to one teacher - student ratio for instruction in the skills area. While sixteen student or half of the class are at the terminals, the remaining sixteen students are in the skill area to receive the minimum of three minutes of individual instruction and attention from the classroom teacher or the resource teacher. Students at the terminals have the proctor for help and attention as needed.

Reports of student progress and placement are available in the lab for cross reference work by the teacher to identify if remediation is needed. These reports also help the teacher know the exact placement of students for new concept work. In like manner, the progress reports, manipulative devices, resource materials, and worksheets can also be integrated into the regular instruction carried out in the classroom on non lab days. The resource teachers during their prep period can consult with other teachers on the staff about student progress in the lab, and help the continuing work of the participating teachers in the lab. This maneuver appeared to lessen the non participating teacher resistance to the CAI pupil and did more

to integrate material than any other process tried. Several handouts of sample math problems and worksheets which paralleled the CAI curriculum were designed and distributed in worksheet form for extra student work at home or other study areas. Many parents who visited the labs became parent volunteers in the CAI instructional system through the worksheets completion at home process.

Four other student assignment systems were tried at one time or another, dependent on the school situation, (1) pullout--a group of three or four classes would send in four or five students to the lab for 20 minutes and repeat the process at the middle of the forty-five minute period, (2) tutorial--the teacher would send in selected students for extra and specific tutoring by the proctor or resource teacher in the skill area, (3) special assignment--bilingual, special education, orthopedically handicapped youngsters participate in the program at special times allotted for them, (4) after school--before and after school sessions are available for those youngsters who need to make up work or by their own interest came in for extra work or help in the CAI lab.

One of the most effective tools for measuring and reporting student progress to the teachers, counselors, and parents was derived by student gain reports on an individual student-class folder. Later, an excellent growth pattern report was developed by graphing the student progress on a tinplate of the math strand ranges. This idea was most positive and very well accepted by parents and teachers, as they could now visually see a graphic record for individual progress through the fourteen strands. The written record sheet which was used by the

proctor and resource teacher to inform the teacher of needed work, location of material, or fixed strand assignments was still kept in the individual student class folder.

The resource teacher and participating teachers feel that "learning takes place in the skill area not at the terminal" because of this new innovative approach to remediation integrates the classroom with the laboratory and provides the teachers and the lab with ample time to plan and use materials at hand.

Integration of Curriculum with CAI

This one subject was labeled "the toughest nut to crack" by the CAI staff. On visitation trips during the pre-implementation staff training, it was noted that most of the CAI labs visited operated as a separate identity. That this separation not be allowed to happen was the one factor which was borne out to the CAI staffs by their CAI coordinator. Initial resistance to CAI from the participating teachers was lessened as they became involved in the planning and implementation process. However, the use of the regular classroom texts, worksheets, prepared learning packages in the CAI skills area created a problem as related literature, models, worksheet or samples were not available for participating CAI staff to use as a guide. Informal meetings between participating teachers, CAI resource teachers, and the CAI coordinator became common place and an avenue for CAI information dissemination. An "on the job" search and evaluation of related materials was instigated. The regular CAI staff meeting, every two weeks, adopted a show and tell structure for an exchange of ideas that had worked, production of sample problems, indexing of

classroom materials and production of worksheets for the participating teachers. It was instantly realized that a summer workshop would be required to find, produce, and index CAI and classroom curriculum materials. To this end, all suggestions, work aids, etc. were kept on file by the resource teachers and CAI coordinator for the assembly of a CAI handbook for the middle school teachers.

September CAI Lab Orientation Procedure

During the first week of school, the CAI resource teachers, CAI coordinator, and proctors met and arranged for the orientation of the participating teachers and students. Copies of the CAI handbook, The Computer Commuter were readied for distribution and the CCC-17 system was checked out for operation. Also during the first week, while school class assignment schedules were being finalized and classes balanced to the proper size, the resource teachers met with their individual principals and received the schedule of classes to be CAI participants.

During the second week, the resource teacher and proctor visited each of the assigned classrooms to meet the teachers and the class. They explained the CAI program aided the participating teachers in orientation of their students. This study orientation on what to expect, the keyboard, reports, etc., proved later to be an excellent nameuver. During this period, the proctor and resource teacher confirmed student names, teacher recommendations, and test scores for proper student placement on the CCC-17 system. The teachers also had a chance to exchange ideas, ask questions, and make recommendations based on the summer workshop while introducing the CAI handbook and

removing any fears of the CAI drill and practice or of the use of the CAI skills area.

On Friday of the second week, classes started to use the CAI lab on demonstration accounts. Procedure and orientation went smoothly with only few problems, created mostly by those students who had not been in school during the first two weeks. Student interest was high and teacher participation in the inservice was almost 100 percent.

Organization of the CAI Middle School Laboratory.

Staffing. Each middle school site is staffed with a resource teacher and proctor. Students are assigned to the lab from regularly scheduled classes and reported to the lab site for their lessons on a regular basis throughout the year.

Responsibilities. Responsibilities of the CAI resource teacher are to:

1. Schedule mathematics classes into the lab.
2. Assist teachers in classroom management techniques and organization using CAI.
3. Recommend teaching strategies and techniques to participating teachers that will aid student progress.
4. Assist in pre and post assessment activities.
5. Assist in the administration of standardized testing as required by the project to gather data.
6. Work with students enrolled in the after school program.
7. Aid in developing data reporting techniques for parents, teachers, counselors, and students.
8. Provide instruction to parents and visitors on the general operation of the CAI program.
9. Develop motivation systems for student participation, rewards, etc.

10. Be responsible to the school site principal and CAI coordinator.

Responsibilities of the CAI proctor are to:

1. Supervise students in the drill and practice area as they operate the computer terminals.
2. Use computer routines to update classes, delete students and produce CAI student reports.
3. Update records with current data of each individual CAI student.
4. Reproduce work sheets and study materials for student use as required by the resource teacher.
5. Program students into the computer system as instructed by the resource teacher.
6. Supervise the student motivation system as established, issuing rewards, etc. as required.
7. Explain the functions of the computer instructional system to visitors and parents.
8. Perform minimal maintenance on equipment as required.
9. Supervise and assign duties to student service in the drill and practice area.
10. Record scores (ID sessions) on individual student record forms and assist resource teacher as assigned.

Responsibilities of the participating CAI teacher are:

1. If the CCC-17 system is down (not operating), the responsibility for instruction of the participating class will return to the regular classroom teacher.
2. Classes are to report to the CAI lab at the beginning of the period on assigned days unless otherwise notified.
3. Participate in CAI inservice and orientation on CAI procedures.
4. When a class is scheduled into the computer lab, it will be the classroom teacher's continued responsibility to:
 - a. Take roll.
 - b. Discipline students.

- c. Interpret student and class reports.
 - d. Prescribe instruction for students using the skills as well as the drill and practice areas.
5. When only part of the class is assigned to the CAI lab: coordinate with resource teacher or proctor to enable:
- a. Students will start immediately on lesson.
 - b. Students are to return to the regular classroom at the end of the lesson.
 - c. Part 4 above.

TEACHER CHECKLIST

The following is a checklist indicating the events you should follow when your students initially enter the CAI program:

- _____ Meet class in September
- _____ Orient class to classroom/CAI lab procedures (Design)
- _____ Submit alphabetical list of class members to CAI resource teacher
- _____ Receive class list from resource teacher indicating student numbers
- _____ Receive initial class placement report from resource teacher
- _____ Use visuals and/or handouts to familiarize students to a variety of problems and how to correctly answer the problem
- _____ Stress expected behavior of students in the lab
- _____ Report to lab on assigned day
- _____ Seat class in assigned area in CAI lab
- _____ Break class into two groups
- _____ Allow 1 group to go to terminals
- _____ Group I receives orientation on keyboard and how to use demonstration account
- _____ Group II receives instruction from teacher on learning materials
- _____ Rotate or switch groups
- _____ Resource teacher orients students on the use of demonstration lessons
- _____ Reinforce and review procedures in classroom prior to third lab visit
- _____ Issue four digit student number
- _____ Review procedure to gain access to system with four digit number

- _____ Students begin to use their number to gain access to their program
- _____ Continue to review lab procedures in your classroom
- _____ Begin to accumulate a data file on each student for each CAI lab
- _____ Receive from CAI lab, class reports
- _____ Post tenth session on student's data sheet
- _____ Post scores on data sheet at regular intervals, keeping track of student's weakness in each strand
- _____ When weakness is noted, begin to work with student to overcome weakness. (Strand level sheets or fixed drill)

Conference Presentation:
"The Courseware Crisis: Help is on the Way"
Judith B. Edwards, Director
Computer Technology Program
Northwest Regional
Educational Lab.
Portland, Oregon
97221

Reprinted with permission of

AEDS

MicroSIFT - A Clearinghouse for Microcomputing in Education

Donald C. Holznagel*

It is not news to observe that microcomputers are being acquired in increasing numbers by schools. Over the last decade there has been a high level of funding and development to support the great potential computers have repeatedly demonstrated in the instructional environment. In spite of these factors, a majority of schools have not taken advantage of the many possible applications of computers.

A 1979 survey of schools in the northwest states, done by the Northwest Regional Educational Laboratory (NWREL), identified a crucial obstacle to wider use: lack of adequate materials. A lack of documentation, validation and dissemination have been cited as factors contributing to the apparent scarcity. This lack of adequate materials is found in both administrative or instructional computing.

Other often cited hindrances to significant use of computers in education are: a lack of skilled personnel to provide materials and training; lack of information concerning the use of computers in curricular areas; technical problems related to hardware/software incompatibility; and, cost.

A clearinghouse for microcomputer instructional software and teacher information has been established at NWREL. The organization is administered by NWREL under a contract with the National Institute of Education. The clearinghouse, MicroSIFT (Microcomputer Software and Information For Teachers), is a central project of NWREL's Computer Technology Program. The program has been productively involved in developing methods and materials for the use of computers in education for more than a decade.

The focus of the MicroSIFT project is on instructional applications for levels K-12. The project will place major emphasis on establishing effective procedures for the collection, evaluation and dissemination of materials and information. Also, technical assistance will be provided to schools in the six-state Northwest region.

The key objectives of the MicroSIFT clearinghouse are:

1. To develop and implement a model for the dissemination of microcomputer software, information, and materials for educational use at K-12 levels.
2. To develop, validate, and implement an evaluation model suitable for computer-based instructional packages. The model will include support of the long-term assessment of the quality of materials proposed for distribution.
3. To develop and implement a user support and technical assistance program in the Northwest region. The purpose of this program is to increase capability for local and state education agencies to choose and implement

"focus of the MicroSIFT project is on instructional applications for levels K-12."

microcomputer applications in education.

4. To develop and implement a feedforward model for guiding and directing development of new computer-based instructional materials.

The MicroSIFT clearinghouse procedures and services have been designed by Dr. Judith Edwards, Director of the Computer Technology Program and Donald Holznagel, Coordinator of MicroSIFT, with significant input from the staff of CONDUIT, an Iowa-based clearinghouse for software for higher education, two MicroSIFT advisory boards (a National Technical Advisory Board and a Regional Users Advisory Board), and other consultants.

It is encouraging to note that many individuals, small and large publishing companies, and hardware manufacturers are beginning to produce programs and materials for

microcomputer applications. However, a primary need of the educational consumer surveying those products is a basis for assessing educational value and utility. The basis for good evaluation of materials is the existence of a well defined process supported by usable instruments validated for the environment of their intended use. A primary concern of MicroSIFT to date, in line with the second objective, has been the development of such a process and instruments uniquely suited to microcomputer-based instructional materials. MicroSIFT staff were joined in this effort by the Program for Research on Evaluation of NWREL, which is exploring new evaluation techniques and environments. Dr. Gabriel Della-Piana of the University of Utah is a principal external consultant in the development of the criteria and the model for evaluating computer based instructional materials.

Sample instruments (checklists, questionnaires, rating scales) used by schools, regional service centers, microcomputer publications and individuals were collected and analyzed. The CONDUIT evaluation procedure and instruments weighed heavily in the initial design. Discussions with organizations having significant experience in the instructional application of computers in the K-12 environment also contributed substantially to the development of this evaluation model.

The resulting evaluation model is in its third version. Version 1 was pilot-tested using 18 programs from a variety of non-commercial sources by a group of six teachers from the Portland Public Schools in Oregon. Three people who are full-time professionals in instructional computer applications were chosen from the Portland area to serve as summary reviewers. Version 2, based on that experience, was tested using over 50 commercial and non-commercial programs by a group of eight teachers in the Beaverton Public Schools in Oregon. Version 3 is being circulated for review and comment to evaluation personnel and experienced users of microcomputer materials.

Training of reviewers in the test groups was done by verbal instruction. An *Evaluator's Guide* is now in

*Coordinator MicroSIFT, NWREL
710 SW Second Ave.
Portland, OR 97204

development based on Version 3 of the evaluation instruments, and will be used on a pilot basis during the 1980-81 school year. The *Guide* will be used in training and as a reference for evaluators to assure accuracy and consistency of procedures and judgments.

The evaluation procedure presently consists of four stages.

1. *Sifting.* An initial judgment is made whether a program or package is suitable for the K-12 microcomputer environment and is ready for evaluation. Certain fundamental criteria must be met: e.g. does it have instructional value (as opposed to strictly recreational)? Can it be run on a microcomputer with little or no adaptation? This step is to be completed by MicroSIFT staff.
2. *Description.* Factual information about the program and materials is reviewed; e.g., instructional purpose, technique, package

"The MicroSIFT Network will be the major vehicle for the review process and the source of reviewers and test sites."

form, documentation available. This information is collected and recorded by MicroSIFT staff and by the reviewers.

3. *Evaluation.* Two or more professional teachers with experience in the program's subject area and stated grade level review the material, test the program, and make judgments concerning its value as an instructional package. A summary review is completed by a person with experience in development and evaluation of computer-based instructional materials.
4. *In-Depth Evaluation.* When a package successfully passes the first three stages and is rated high enough to warrant further evaluation, instructional effectiveness will be assessed through pre-testing and post-testing, student observation, or a combination of these methods. An expert in the subject matter content of the package will supervise the In-Depth Evaluation. This person will make a recommendation to MicroSIFT regarding the usefulness and effectiveness of the program.

The instruments and procedures for Stage 4 of the process, aside from subject matter test instruments, are still in design and development. This

stage was not pilot tested by the two test groups. It is planned that several alternatives for Stage 4 would be available to evaluators as indicated in the description above. Dr. Della-Piana is now working on instruments and guidelines for the observation of student use of a microcomputer-based instructional package. Pilot testing for these instruments will be completed during the 1980-81 school year.

Established educational institutions and consortia will be the major initial mechanism for the collection, evaluation and dissemination activities of MicroSIFT. A network of organizations is being established consisting of existing computer centers which serve a large number of schools and provide significant work in instructional applications of computers. Their personnel are experienced in computer-based education and in dealing with the technical problems involved in program transfer. Such centers have a need to evaluate the efficacy of instructional software. Many have done evaluations on a small scale, but are not able to support large-scale continuous evaluation efforts. They will serve as long-term sources of materials and information. In addition, these centers will serve as agents for review, product documentation and finishing, field test sites, and further development.

The Network will be established in two stages. In the first stage a selected set of centers will be enlisted as initial members. The second stage of the Network will be based on successful activities in the first stage. Alternatives for the second stage will be discussed at meetings of Network participants at which the collection and evaluation processes are reviewed.

The MicroSIFT Network will be the major vehicle for the review process and the source of reviewers and test sites. The centers offer access to experienced and beginning users, qualified summary reviewers, a variety of student groups, geographic locations, and ethnic, minority, and socioeconomic groups. They are in large part experienced in evaluation of computer-based materials. Almost any desired hardware base can be found in use among them. A field test of the evaluation process, instruments and *Evaluator's Guide*, will be conducted at Network sites during the period from November through January, with resulting reviews published in February, 1981. Additional field test activities will be conducted in a university course environment by Dr. Robert Woolley, Utah State University, Logan, Utah.

Information concerning clearing-house plans, activities, progress, services, and outputs will be made

available through several avenues. One of the primary avenues is the expanding general dissemination capacity being directly supported by NIE: Regional Exchanges, Regional Service Programs, and State Capacity — Building Programs. The *CEDaR Comments and Educational R & D Report*, publications of the Council for Educational Development and Research, will be used for spread of information to the readers of those publications. A MicroSIFT Newsletter has been established to facilitate spread of information through direct mail contact. A continually growing mailing list is being compiled. The list of interested people, primarily from the United States, also includes names of people from other countries. The newsletter will be produced quarterly beginning in October, 1980. News releases and articles will be prepared for publication in professional magazines which are specifically aimed at the educational computing community.

Bibliography

"Educational Computing in the Northwest, 1979: Status and Needs for Information and Assistance," Northwest Regional Educational Laboratory, June, 1979.

"Design and Operation of MicroSIFT," Northwest Regional Educational Laboratory, August, 1980.

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

For the year ending 12/31/80

1. Name of publication: **MicroSIFT**

2. Issue frequency: **Quarterly**

3. Issue date for circulation data below: **12/31/80**

4. Annual subscription price: **\$5.00**

5. Number of copies of this issue published: **1000**

6. Total number of copies of this issue: **1000**

7. Total number of copies of all issues published during the year: **4000**

8. Total number of copies of all issues: **4000**

9. Total number of copies of all issues: **4000**

10. Total number of copies of all issues: **4000**

11. Total number of copies of all issues: **4000**

12. Total number of copies of all issues: **4000**

13. Total number of copies of all issues: **4000**

14. Total number of copies of all issues: **4000**

15. Total number of copies of all issues: **4000**

16. Total number of copies of all issues: **4000**

17. Total number of copies of all issues: **4000**

18. Total number of copies of all issues: **4000**

19. Total number of copies of all issues: **4000**

20. Total number of copies of all issues: **4000**

21. Total number of copies of all issues: **4000**

22. Total number of copies of all issues: **4000**

23. Total number of copies of all issues: **4000**

24. Total number of copies of all issues: **4000**

25. Total number of copies of all issues: **4000**

26. Total number of copies of all issues: **4000**

27. Total number of copies of all issues: **4000**

28. Total number of copies of all issues: **4000**

29. Total number of copies of all issues: **4000**

30. Total number of copies of all issues: **4000**

31. Total number of copies of all issues: **4000**

32. Total number of copies of all issues: **4000**

33. Total number of copies of all issues: **4000**

34. Total number of copies of all issues: **4000**

35. Total number of copies of all issues: **4000**

36. Total number of copies of all issues: **4000**

37. Total number of copies of all issues: **4000**

38. Total number of copies of all issues: **4000**

39. Total number of copies of all issues: **4000**

40. Total number of copies of all issues: **4000**

41. Total number of copies of all issues: **4000**

42. Total number of copies of all issues: **4000**

43. Total number of copies of all issues: **4000**

44. Total number of copies of all issues: **4000**

45. Total number of copies of all issues: **4000**

46. Total number of copies of all issues: **4000**

47. Total number of copies of all issues: **4000**

48. Total number of copies of all issues: **4000**

49. Total number of copies of all issues: **4000**

50. Total number of copies of all issues: **4000**

51. Total number of copies of all issues: **4000**

52. Total number of copies of all issues: **4000**

53. Total number of copies of all issues: **4000**

54. Total number of copies of all issues: **4000**

55. Total number of copies of all issues: **4000**

56. Total number of copies of all issues: **4000**

57. Total number of copies of all issues: **4000**

58. Total number of copies of all issues: **4000**

59. Total number of copies of all issues: **4000**

60. Total number of copies of all issues: **4000**

61. Total number of copies of all issues: **4000**

62. Total number of copies of all issues: **4000**

63. Total number of copies of all issues: **4000**

64. Total number of copies of all issues: **4000**

65. Total number of copies of all issues: **4000**

66. Total number of copies of all issues: **4000**

67. Total number of copies of all issues: **4000**

68. Total number of copies of all issues: **4000**

69. Total number of copies of all issues: **4000**

70. Total number of copies of all issues: **4000**

71. Total number of copies of all issues: **4000**

72. Total number of copies of all issues: **4000**

73. Total number of copies of all issues: **4000**

74. Total number of copies of all issues: **4000**

75. Total number of copies of all issues: **4000**

76. Total number of copies of all issues: **4000**

77. Total number of copies of all issues: **4000**

78. Total number of copies of all issues: **4000**

79. Total number of copies of all issues: **4000**

80. Total number of copies of all issues: **4000**

81. Total number of copies of all issues: **4000**

82. Total number of copies of all issues: **4000**

83. Total number of copies of all issues: **4000**

84. Total number of copies of all issues: **4000**

85. Total number of copies of all issues: **4000**

86. Total number of copies of all issues: **4000**

87. Total number of copies of all issues: **4000**

88. Total number of copies of all issues: **4000**

89. Total number of copies of all issues: **4000**

90. Total number of copies of all issues: **4000**

91. Total number of copies of all issues: **4000**

92. Total number of copies of all issues: **4000**

93. Total number of copies of all issues: **4000**

94. Total number of copies of all issues: **4000**

95. Total number of copies of all issues: **4000**

96. Total number of copies of all issues: **4000**

97. Total number of copies of all issues: **4000**

98. Total number of copies of all issues: **4000**

99. Total number of copies of all issues: **4000**

100. Total number of copies of all issues: **4000**

NETWORK SYSTEMS FOR SCHOOLS

Cassie C. Connell
CORUS SYSTEMS, INC.
2029 O'Toole Avenue
San Jose, CA 95131

I. REQUIREMENTS OF A NETWORK

A. The Network

A network means that more than one of something has been connected to the others of its kind, usually because of an economic incentive or advantage.

That is true of local television stations that are affiliates of the large shared medium, the major network, and it is true of a group of computers that share the large storage medium the CORVUS disk drive, and the common peripherals.

B. Computers

There are several major microcomputers on today's market. Among these are APPLE, the TRS-80 by RADIO SHACK, ALTOS, ALPHA MICRO, EXIDY, SUPERBRAIN and NORTHSTAR. Some of these computers use an operating system that is written in BASIC, some in CP/M, and some in UCSD PASCAL. CORVUS has the capability of putting any or all of the operating systems on the same network. The computers are able to talk with each other and transfer files, via pipes, from one OS to another OS. For example, a file in CP/M can be sent through the pipe to another computer. That computer might use UCSD PASCAL as its OS. It can accept the file and send it to the disk in PASCAL. In purchasing the computers, consider the stability of the company, reliability of the microcomputer product and service availability.

C. Storage

Buy as much storage as the budget allows. One can never have too much storage. The CORVUS Winchester DISK DRIVE is available in either 10 or 20 megabytes and permits daisy chaining up to 80 megabytes on line. If more than one OS is in use on-line, the disk is divided into blocks for use by each operating system.

D. A Means of Connecting the Computers

The CORVUS CONSTELLATION is a multiplexer. There are two kinds of multiplexers, a master and a host. A master mux can connect either host multiplexers or computers to the DISK DRIVE. A host multiplexer connects computers to the DISK DRIVE or onto the master multiplexer which, in turn, is connected to the DISK DRIVE.

The network permits each computer to:

- (1). Share the large data storage medium (the CORVUS DISK DRIVE)

- (2). Share any common peripheral, like a printer.

The cost of a Diablo letter quality printer is between \$2910 & \$1400. It is economically advantageous for all the computers to be able to use one printer. The software that is part of the CORVUS CONSTELLATION and disk drive makes this possible.

- (3). Communicate with other computers on the network.

- (4). Transfer a file from a computer with one OS to a computer with a different OS.

E. BACKUP

Every computer user knows you should backup your data at the end of every working day. The CORVUS MIRROR is an efficient, fast, and economical method for backup. The MIRROR is used to transfer the data from the disk drive to any standard VTR (VCR). Here's what happens. The mirror takes the digital signals from the disk drive, changes them into video signals, and sends them to the VTR (VCR) for storage. This transfer occurs at the rate of one megabyte per minute; a two-hour tape holds 120 megabytes of data.

II. HOW THE CORVUS NETWORK WORKS

(A). Traditional Backend Network (See Figure 1)

Originally, this network for large computers was for communications only. Each large computer had its own storage and peripherals. Today, each computer has access to the mass central storage and shares common peripherals, and communicates with the other computers.

(B). Star Backend Network (See Figure 2)

The Star configuration is used for microcomputers. In a star network the structure surrounds the central storage and gives the microcomputer direct access to the large shared data medium. The central storage is accessed directly by each computer without going through another computer.

(C). The CORVUS CONSTELLATION

Host computers A, B, C, D, and E are five of the possible fifteen computers that could be connected to the constellation. No dedicated computer is required. Each computer directly accesses the disk as though it were the only one connected. Here is a configuration for a total of sixty-four computers.

(D). The System Controller

A Z-80 microprocessor, the system controller, resides in the disk drive and contains 16 kilobytes of RANDOM ACCESS MEMORY. (See Figure 4) This controller uses DIRECT MEMORY ACCESS hardware to control the disk drive, the backup device, and the host multiplexers. It is this microprocessor that controls the data protection (security) and communication facilities.

(E). BACKUP

Every computer should be backed up at the end of every working day. CORVUS recommends the CORVUS MIRROR. The MIRROR is an interface that translates the digital signals from the disk drive into video signals for storage on any standard VTR (VCR). This form of backup is low in cost, reliable, and fast. The backup occurs at the rate of one megabyte per minute. The VTR may be VHS or Betamax and the transfer occurs in any one of the three formats, NTSC, PAL or SECAM. A 2-hour video tape will hold 120 megabytes.

(F). Software

The CORVUS software is written to prevent any one computer on the network from using the same area of the disk that another computer on the network is using.

The CONSTELLATION allows each computer to use its own native operating system- and without modification. Probably, the most commonly used operating systems among microcomputers are BASIC, CP/M, and UCSD PASCAL. There is a quantity of tested software available for these systems. Many Computer Users In Education (CUE) groups have software sharing or swap groups, and it is from these groups that one may obtain valid critiques on the software. Among these educational user groups are:

- (1). MECC, St. Paul, Minneapolis
- (2). ADCIS, Assistant for the Development of Computer Based Instructional Systems, Billingham, Washington.
- (3). CUE, in several states.

100

The Operating System of each computer interfaces to the disk drive through the interface card placed in the computer. A cable runs from the card to the multiplexer. Another cable runs from the multiplexer to the disk drive. Through this I/O medium each computer accesses the disk drive. A unique drive, (slice of software that goes between the O/S and the peripherals) acts as the I/O to the disk drive. The Constellation contains a drive (the software) for each supported system. The I/O drive appears to the operating system as a system disk.

Each area of the disk is allotted x-number of blocks (a block is 512 bytes) and this allotted area is called a VOLUME. Each VOLUME contains a DIRECTORY of the FILES in that volume. (See Figure 5)

Each user can be assigned any one of, a combination of, or all of the following for each volume on the disk drive.

- (1). READ ONLY- which means that the user can access any file in the volume and read it but can neither type anything into, nor remove anything from the file.
- (2). READ WRITE-which means that the user can read any file in the volume and either type in information or erase information from the file.
- (3). NO ACCESS- which means that the volume is invisible to the user and cannot be accessed.

This assignment to the mount table is made by the system manager when the system is setup initially. Of course it can be changed by the system manager at any time.

The volumes that are stored on the disk are either-

- (1). system- can be read by the users but cannot be written to by the users. This "system" VOLUME typically contains all the common programs and utilities (e.g. text editors, compilers). It is possible to make this a "no access" volume for all but the systems manager.
- (2). personal-can be accessed by only those users assigned to it by the mount table.
- (3). shared- all users have read/write access to shared volumes.

It is the shared volume that requires the constant monitoring by the system. Because all computers have access to shared areas

it is necessary to control the access. CORVUS does this through the use of semaphores. A CORVUS semaphore is an eight character label. Corvus provides these eight-character names for a file, volume, block, or sectors. The user software defines what is protected by semaphores. It is user defined. If the user sets up files with semaphores and one user asks for a file that is in use at another computer, the access is denied to the second requestor.

A file in use sets up an "in use" semaphore, the eight character name, and prohibits any other entry into the file. Thus, two computers cannot access a shared file at the same time.

G. DATA SECURITY

Corvus offers data security through the assignments made to the mount table. These assignments are made by the system manager and the system manager is the only one who can later change these assignments. The system manager determines what user has access to what volumes, and sets up a file that lists who has access. When the computer boots, a log-in program checks the user's identity and admits him to only those volumes identified as available to him according to what was set up in the mount table. This data security scheme is available for several operating systems, including APPLE BASIC, CP/M, and UCSD PASCAL.

H. COMMUNICATION

The CONSTELLATION, see Figure 6, provides a means for any microcomputer to communicate with any other microcomputer on the network. The CONSTELLATION permits communication through data paths that are called "pipes". The pipe functions as a transmission line for information sent serially byte by byte. The pipe functions on a FIFO basis, the first piece of information in is the first piece of information out at the other end. The pipe resides in an assigned area of the disk and is managed by the system controller.

These pipes also make it possible to transfer text files from a computer with one operating system to a computer with a different operating system. Because of these pipes it is not necessary for one operating system to understand the other operating system. The transfer occurs in the form of sectors. The user enters a command that activates a utility program. The user can then send a text file written in PASCAL to the PIPE to be held there until it is requested. The user of a CP/M system on the network can then ask for the text file and it is sent from the pipe to his computer. From this computer the user sends the file to the disk drive. This time it goes to

the disk drive in the language of this operating system, the CP/M language.

The size of the pipe(s) is determined and specified when the system manager runs the DIAGNOSTICS program at initialization.

SPOOLING

Spooling is defined as a technique whereby output to a slow device (printer) is placed in a queue on a mass storage device to await transmission to the slow device (printer).

The term "spooling" came into use when computers were young and disks were large and held minimal amounts of data. At that time data to be printed was transferred to a roll of magnetic tape which was on a spool. The practice began to be known as "spooling". The purpose of "spooling" was to permit the computer to work on other jobs. Information was held on the spool until a slower time period when the data could be printed.

In the 60's greater capacity disk storage permitted the disks to hold the data in a queue on the disk until time for printing. However, the term spooling continues to be used and is now defined as "simultaneous peripheral operation on-line."

Today, through the software available on CORVUS' CONSTELLATION, spooling is available for microcomputers on the network.

Each microcomputer can stack print jobs in a queue. When it is time to print, one computer is dedicated to printing, and it accepts the print commands on a FIFO basis and prints all that is in the queue. This is the only time that a dedicated computer is required.

III. How to Begin With One Computer and Build A Network from There

It can be done. If you have one computer, let's say it's an S-100 Bus and uses CP/M as the language of its operating system, and you need the capability and flexibility of a mass storage medium, a CORVUS disk drive in either 10 or 20 megabytes should be the next investment.

In the next budget allot a sum of money for the CONSTELLATION (the multiplexer that permits networking), two computers, and a printer. Now the system has three interactive computers on the network and a printer. The CORVUS CONSTELLATION networks the computers, allows each one access to the disk drive, and the printer, and permits the computers to communicate with each other. On the presumption that these last two computers were APPLES, the system now contains computers that use BASIC as the language for their operating system. This configuration works on the CORVUS network. Remember that if you already have several computers, and they are not the same brand and do not use the same operating system, they all will function on the CORVUS network.

When two or more languages are in use on the network, the individual who initializes the system divides the disk into areas for use by each computer. In the preceding case, part of the disk will be available for CP/M and part for BASIC.

Through the use of pipes, it is possible to transfer text files from a computer with an operating system in the CP/M language to the computer with an O/S in BASIC.

From this point, the budget should be set up each year to add computers and constellations to the network. How big and over what period of time is dependent upon funding.

Perhaps it is more realistic for you as educators to look at a four year plan with a major purchase each year.

In your folder is a page titled CORVUS SYSTEMS, INC.- A NETWORK. Let's examine that page first.

The next page provides a breakdown in the cost of the CORVUS 20 Megabyte disk drive versus the Floppy drives. After adding the costs for each over a four year period, the totals come to \$18,475 for the hard disk and \$41,440 for the floppy drives. On the basis of purchasing used here, there will be eight computers available, then 16, then 24,

and finally 32. If one calculates from the premise that there are five class periods per day and the number of students per class equals the number of computers available per year, then at the end of four years, 400 students will have been through the classes. If one divides 400 into the totals, the cost per student is

\$46.54 if CORVUS is used

\$103.60 if floppy drives are used

The floppy cost more than doubles the cost per student.

Cost is not the only factor. The advantages of the CONSTELLATION and the CORVUS DISK DRIVE over floppy drives are numerous.

From the teacher's point of view....

1. The teacher has the capability of accessing any student's file at any time.
2. The teacher can check a student, from the teacher's computer, any time during class to monitor work in progress.
3. The teacher can put a lesson or a message on the disk drive and each student can access that message or lesson.
4. The student has access to only his file -- part of the data security program -- and cannot enter another student's file.
5. The teacher can keep grade records, using programs that work through the percentages for the teacher. The teacher is able to quickly and accurately produce grades for any student at any time.
6. Attendance records and reporting are much easier with a computer to assist the teacher.

From the administrator's point of view, the large storage medium, available from the beginning, makes it possible to use the system for student record keeping; for equipment, furniture, and book inventory; for budgeting; for payroll; for club accounting; for preparing bids; for scheduling building use either for the community or school itself. The list is endless. These are merely examples of how the computer can aid both the administrator and the teacher.

Record keeping on a computer is fast, efficient, and cost effective.

The last page in the file lists the four-year cost of a system that uses floppy drives.

The cost of the computer is the same with either the hard disk or the floppy systems.

The cost of a floppy system is nearly \$23,000 more than the cost of a more efficient networking system that uses the CORVUS Disk Drive.

The floppy system does not permit:

1. networking
2. a mix of computers
3. central access to all areas of work by the teacher
4. a queue for printing
5. communication between computers
6. the use of common peripherals
7. fast efficient backup
8. ease of operation
9. data security

Networking is the most cost efficient method. It is also the most advantageous use of the computers.

Mr. Linn Boswell
Math Department
Desert Shadows Middle School
Paradise Valley School Dist.
866-1194

1,000 STUDENTS, ONE COMPUTER - PROBLEMS!

To help eliminate problems, the following should be considered:

I. Getting the computer -

1. Why a computer?
2. Why only one?
3. Your needs?
 - a. Color
 - b. Sound
 - c. TV/Monitor
 - d. How much memory
 - e. Which peripherals

II. In the school -

1. Where does it go?
2. Who is in charge?
 - a. You
 - b. Math department
 - c. Principal
 - d. Media center
3. Who gets to use it?
4. How will it be used?
 - a. Teaching programming
 - b. Computer assisted instruction (CAI)
 - c. Administrative purposes
 - d. Multi-purpose
5. How should it be scheduled?
6. Maintenance?

Programs Guaranteed Not to Teach
(Recommended Computer Applications for Remedial Reading Centers)
H. Donald Jacobs, D.Ed.
Reading Clinic Coordinator
Arizona State University

The purpose of this presentation is to share with you the observations and objectives that are guiding my colleagues and myself. Hopefully, it may help you avoid some of the frustrations that others have suffered. I will first describe two of my professional activities so that you will understand my perspectives and better appreciate our objectives.

I come from the world of reading instruction. More specifically, my specialty is remedial reading. Thus, my knowledge and perspectives of the micro-computer area are narrow in terms of curriculum.

In the past four years, the majority of my consultations have been concerned with evaluation, re-organization, or administration of remedial reading programs and resources for better efficiency; not for more effectiveness, nor for developing more instructional materials. In other words, to improve the support system for the teacher. The following observations are from those activities, or perspectives, and are the foundation for our efforts.

The teacher is the most critical, or significant, component within the instructional setting. Teacher characteristics, as a composite, more significantly influence whether reading achievement will occur than do any other classroom factors, including student characteristics. The teacher's interpersonal characteristics seem to become even more significant in the remedial reading domain. The degree of negative attitudes among students in remedial reading is notably greater than among those in the developmental reading program. This problem must be addressed, directly and inter-personally, if any residual remediation is to occur.

The record-keeping demands upon remedial reading teachers in the last few years seem to have increased logarithmically and with no end in sight. This is a result of federal monitoring; state monitoring, such as CUES; local requirements, such as IEP's, criterion management instruction, and minimal proficiency programs. These documentation demands have reduced the available teacher-time to the student. This problem is exacerbated when teacher-aides are prohibited from performing clerical duties. The teachers must assume the clerical support responsibilities for the aides as well as for themselves. The result is that the most significant component, the teacher, which the vast majority of remedial reading students need most is being withdrawn.

The high frequency in the late 1960's of sabotage to teaching machines in remedial reading labs with accompanying repair and security costs contributed, if not lead, to their demise (undeserved in many cases) in many districts, to my knowledge. I believe this is related to the inter-personal importance of the teacher. In any case, it definitely should cause some caution as to the extent of instruction that should be computer based. The probability of 100% machine-based instruction is much less now than at that time, ~~due to the costs~~. I wish I could say, with assurance, that it is due to improved educational knowledge and judgement, *not due to costs*

Remedial reading teachers have an increasing need for organizational support and efficiency: material retrieval by instructional specifications, resource allocation decisions, diagnostic and achievement data tabulation, and routine lesson preparation procedures. This increase is due to the increases of available materials, diagnostic data, variety of remedial students, and the decrease of the teacher's available preparation time.

Greater sophistication or complexity of teacher-support computer programs eventually have an inverse effect upon teachers' utilization. When the teacher is forced to spend what is viewed as an excessive amount of time at the terminal entering information, utilization will drop. The same is true when forms must be completed by the teacher hours or days prior to their need, for a terminal operator to process.

Too often, program designs have been primarily influenced by reading specialists or computer specialists. The results can be retrieval descriptors that don't communicate (ex. encoding vs. decoding, word ends vs. endings) or programming logic for system efficiency that is not congruent to common teacher logic ^{(ex.} for selecting instructional materials^{).} This usually results in operator (teacher) confusion and error. Nothing reduces utilization faster than a teacher getting the message, "ILLEGITIMATE DATA ENTRY," three times in a row during lunch break, or two minutes before the next group of students arrives.

Finally, when budget cuts begin, these support programs tend to be the first sacrifices. The individuals who maintain the programs and the data are usually the "first-to-go." They are replaced by superficial training of other personnel such as the librarian or reading coordinator. But, they do not have sufficient time nor expertise. Eventually, the program is inadequate, then invalid, and finally inoperative.

Three years ago, my colleagues and I decided to see if we could eliminate some of these problems. Our first effort was to establish a goal with subsequent objectives. They are: Put the teacher back on the "classroom floor" for more student-teacher contact; to be achieved by:

1. Select and develop a system that is immediately available to the teacher;

2. Develop programs for retrieval of remedial reading instruction materials and of reading assessments;
3. Develop programs that list student responses to objective assessments rather than just scores, so that the teacher can do error pattern analyses for more valid diagnoses;
4. Develop programs that eliminate/reduce repetitive clerical activities for the teacher;
5. Design all programs to conform to predominant vocabularies and logics of classroom teachers; and
6. Design those programs for dynamic data (such as retrieval bibliographies and student programs) to be maintained by the classroom teachers.

We have not yet fulfilled our objectives but we believe we are making progress. At least, we have not violated any of them. And, it is from these that came our catch-phrase and the title of this presentation, "Programs guaranteed not to teach."

The first objective has been the easiest, the selection of a system. No great effort was necessary to determine that the micro-computer was most appropriate for our intentions. In fact, some of its limitations (in comparison to larger units) now appear to be advantages, for example, ease of operation. Our progress toward the other objectives is best illustrated by the list of our programs and progress descriptions (following).

The ALPHA program is being used by some teachers who use the language experience instructional method. They prepare "before" and "after" word lists from students' language charts to illustrate gains in vocabulary to parents. This is effective when there are gains that do not appear on standardized tests and it is qualitative rather than just quantitative.

The CLOZE procedure is quite prominent in the professional reading literature and is widely advocated for many purposes.

The COMPUTER ASSISTED DIAGNOSIS programs tend to be the most popular notion with reading specialists. Current programs score assessments and print the scores. They do not list the errored items for specific, diagnostic interpretation for which scores have absolutely no predictive validity in individual reading remediation.

The READABILITY programs have applications for reading teachers, librarians, and text adoption committees, primarily. Another use has been the measurement and editing of local district publications.

The RETRIEVAL programs are the most challenging for us. The programming was the easiest. The reliability of a taxonomy and a logic with classroom teachers has been the problem. We are just beginning to believe we are making progress.

ALPHA. This program has been used for pre-/post assessment (reading or writing), inter-text comparability, text-test comparability, and linguistic research. The text is typed in. Output (initiated by an option code) is an alphabetical listing (hard copy) of the different words and their frequencies from the text. This program is being revised to allow a storage option.

CLOZE. This procedure has been used to reinforce use of context clues, to test content achievement, to measure readability, and for linguistic research. The desired deletion frequency, the identification of the passage source, and then the text are typed in. The text may be terminated at any time. Output (hardcopy) includes heading (passage identification and space for the student's name), mutilated text (first and last sentences are untouched, blanks equal 12 spaces) and a teacher-key, which is an identical copy but with the deleted words included and flagged. This program is being adapted for micro-computer application and large system exportability.

COMPUTER ASSISTED DIAGNOSES. These programs are for scoring and error listing individually administered objective response tests. The operator (teacher) enters the number or letter for each of the answer options selected by the student. All options have been previously classified such as: (a) Initial, Medial, Final positions; (b) VCV, VRV, VCCV, etc; (c) C, CC; (d) Phonic, Syllabic, Structural. The program output (hardcopy) is a listing of (a) number correct, (b) number of each type of error, and (c) list of errored items, categories, and correct options. The program reduces time of scoring, tabulation, and preparation of one test for diagnostic analyses from 2 1/2 hours to approximately 18 minutes. These programs will be available upon completion of rights negotiations.

MANUAL. The administrator enters a request for CALENDAR at a terminal. A prototype calendar is produced; this is a listing of tasks to be performed according to each week. Each task is identified by a number. A listing of procedures and a copy of any necessary documents for a particular task are produced when the task number is entered as a request. There is also a secured option (a code entry is required) which requests the entire manual to be printed. Samples of output are available.

READABILITY. The following programs are for estimating the reading difficulty levels of printed material. They are based upon the formulas most prominent in the professional literature. The following features are common to all. The text is typed in. Before each

entry line, the total of running words already entered is printed. Output (initiated by an option code) includes a readability score and totals, frequencies or ratios of the text elements used in computation. Final output is an average score for the samples. A text editing option exists in each program.

READABILITY:DALE/CHALL. This formula is applied to fourth grade through high school material. Below each entry line, "uncommon words" (not on "Core" list) are listed. This program is now being adapted for microcomputer application and prepared for large system exportability.

READABILITY:FLESCH(A). This formula is applied to fourth grade through graduate college material. Computation is based upon average sentence length and a vowel :: syllable algorithm (r .94). This program is undergoing revision.

READABILITY:FLESCH(D). This program differs from FLESCH(A) in its readability computation. Below each entry line, "uncommon words" (not in "Core" list and not in "Dynamic" list) are listed. The operator enters the syllable count for the uncommon word, it and its syllable count become entered in the Dynamic list. Computation is based upon average sentence length and exact syllable count. The Dynamic list may be purged of low frequency words. This program is being field tested.

READABILITY:SPACHE. This formula is applied to primary materials. The program's features are identical to DALE/CHALL; the only difference being the Beta weights in the formula computation. This program is now being adapted for micro-computer application and prepared for large system exportability.

RETRIEVAL. These programs are for prescriptive retrieval of materials. Three separate libraries are envisioned: "Assessment," "Instructional," and "Tradebooks." The operator will enter significant student or material descriptors and a controlled bibliography will be listed. The goals are to have (1) "teacher-priority" taxonomy, (2) simplified taxonomy, (3) faculty maintenance of listings and (4) tolerance of site idiosyncrasies. The "pilot" programming is successful, currently "field-testing" different taxonomies and different update procedures.

R_x For The 80's
Microcomputers And Problem Solving

by

Gary G. Bitter
Arizona State University
Tempe, Arizona

An agenda for action published by the National Council of Teachers of Mathematics recommended that:

1. Problem solving be the focus of school mathematics in the 1980's and
2. Mathematics programs take full advantage of the power of calculators and computers at all grade levels.

Of course there are other recommendations but these two have similar objectives. Problem solving involves strategies, deductive and convergent thinking, logical reasoning and the ability to form models from real world situations. As the California framework so succinctly states, "Mathematics is problem solving." The microcomputer is a means to facilitate problem solving.

In the last few years the microcomputer movement has evolved to play a viable role in education. This movement can be compared to other forms of technology. For example, it took 35 years from discovery of the vacuum tube in the 1880's to the development of a practical radio. The x-ray tube took 18 years to develop, television evolved in 12 years. The nuclear reactor took 10 years, to develop the atomic bomb a mere 6 years. The computer technology growth can be thought of in exponential fashion and is often referred to as a revolution. The first generation computers were characterized by the use of vacuum tubes. The 1946-1959 computers could do about 40,000 additions per second at a cost of \$1.25 per 100,000 computations. The second generation was brought about due to the inven-

tion of the transistor. These 1959-1964 computers could do about 200,000 additions per second. The cost had dropped to 25¢ per 100,000 calculations. The third generation, with the capabilities of 1,250,000 computations, was stimulated by integrated circuits. These 1965-1977 computers could do 100,000 computations for 10¢. Large scale integration was the beginning of the microcomputer, or fourth generation. These machines could do over 10,000,000 additions per second at a cost less than 1¢ per 100,000 computations. Hewlett-Packard announced in February of 1981, the development of a chip that can process the information of 1000 books in a single second. This computer chip has a 450,000 transistor capacity. As can be seen the growth is exponential. From 1950, the capacity per add time in seconds with a computer was around 10^8 , 1960 was 10^{12} , 1970 was 10^{16} and today's models make those seem "turtle slow".

This revolution is commonly characterized with the following analogy. If the same improvements had followed the Rolls Royce you could buy a Rolls Royce for \$2.75. This Rolls Royce would make 3 million miles to a gallon of gasoline and six of them would fit on a pin head. We have amazing technology available at a lower and lower cost. Technology aided instruction is now becoming economically feasible

Microelectronics technology is constantly increasing our ability to communicate, understand complex problems, store and recall large amounts of information and create knowledge.

But there is resistance from consumers and educators. Computers are in the news for such negative events as the near alert in our defense system or some billing problem from a utilities company. In education

they are just another new idea or panacea.

To overcome this resistance we will need to create an "ease of use" or computer literacy. People have to realize that the microcomputer is going to invade all portions of our lives, not only our industries and homes, but our classrooms as well.

Microelectronics development has surpassed our ability to apply it to uses in society. The same is true for education. But the technology is becoming more common in society, where over one million home computers will be in use by 1984. Education is also becoming actively involved. To emphasize this point, Creative Strategies market survey shows that in 1979, approximately 36,000 microcomputers were sold to educational institutions for about \$40 million. In 1980, the number of units sold into this area grew to 72,000 units totaling about \$95 million. It is estimated that by 1985, over 130,000 units will be brought into the educational environment.

A recent study by Chambers and Bork indicates that 90% of the school districts in the United States use a computer and it is expected that this number will increase to 95% by 1985. In addition, instructional usage has increased from 15% in 1970, to 74% by 1980, with 85% to be expected by 1985. Of course the major usage is in Secondary Education with math the main application area.

The concerns of education are several. First, there is a general anxiety toward technology and technology assisted learning. Feelings of fear and tension pervade the first time user.

Hardware and software are concerns. There is worry that the hardware will not consistently work without additional repair. Software availability for specific hardware is also a justifiable concern.

Boredom in computer assisted lessons can be a common problem unless variety and good software are used. Teachers fear that initial interest will be followed by a fading later involvement and learning.

Teacher preparation in the use of the microcomputer and the ability to apply it effectively to curriculum needs is a valid concern. Teachers need to be adequately prepared to understand the microcomputer and its implications in the classroom. Of course there are many other issues which could, and will, be raised in the years ahead.

The factors presently influencing schools toward microcomputer involvement are low cost, individualized applications and curriculum options. Many schools are purchasing home computers on a trial basis. It is important not to be left out of this phenomena. Teachers look upon the microcomputer as an extra helping hand to teach low, average and high ability students. Many are also interested in classroom management options.

In essence, there are three areas of microcomputer application in education. They are computer awareness, computer managed instruction and computer assisted instruction. Computer awareness, or computer literacy, is the ability to identify a computer and to use it reasonably well without having to become an expert. We must stress that the situation should not exist where a person is afraid to walk up to a computer and type in some information.

The following outline represents a quick look at some content which can be included in a computer awareness or literacy discussion.

COMPUTER AWARENESS TOPICS

I. Introduction

A. Computer Capabilities

- a. Science fiction discussion
- b. Actual capabilities (speed, accuracy and repetiveness)

II. History of the Computer

- A. Abacus - 3000 B.C.
- B. Pascal - first mechanical adding machine - 1642
- C. Babbage - mechanical computer - 1830
- D. Hollerith - developed punched cards- U.S. Census - 1900
- E. Analog Computer - 1931
- F. Mark I - first general purpose computer - 1944
- G. ENIAC - 1945
- H. Werner Von Braun - computer technology
- I. IBM - 1959
- J. Cost

III. Computer Generations

- A. Vacuum tube (first) - 1946
- B. Transistor (second) - 1959
- C. Integrated Circuit (third) - 1964
- D. Large Scale Integration (fourth) - 1978
- E. Very Large Scale Integration (fifth)

IV. How a Computer Works

A. Input

- a. Types (terminal, CRT, cards, paper tape)
- b. Accessing a computer

B. Output

- a. Types (terminal, CRT, graphic, paper)

C. Arithmetic Unit

- a. Binary Arithmetic

D. Central Processing Unit (CPU)

E. Memory

- a. Types (disk, magnetic tape, magnetic core)
- b. ROM, RAM

F. Software - Hardware

- a. Languages
- b. Stored program
- c. Parts of a computer

G. Compare Computer to Human Solving $2 + 3$

H. Searching, Sorting, Deleting and Updating

V. Computer Types

- A. Computer Systems
- B. Mini Computer
- C. Micro Computer
- D. Time Sharing Computers
- E. Microprocessors

- a. Uses

VI. Home Computer

- A. Types
- B. Applications

- a. Passive
- b. Active

VII. Computer Uses

A. Society

- a. Banking
- b. Department stores
- c. Medicine
- d. Law enforcement
- e. Government
- f. Weather prediction

B. Education

- a. Computer assisted instruction
- b. Computer managed instruction
- c. Computer extended instruction

C. Data Processing

VIII. Computer Misuses

A. Fraud

- a. Examples (stealing money and information)
- B. Identification codes, pass words and data files
- C. Detecting computer-based crimes
- D. Privacy (credit checks)

IX. What a Computer Can and Cannot Do

- A. What is a Computer
- B. Programming
- C. Flow Charting

X. Computer Careers

- A. Computer Operator
- B. Keypunch or Coding
- C. Programmer
- D. Systems Analyst
- E. Computer Scientist
- F. Artificial Intelligence

XI. Computer Errors

- A. Examples

XII. Future and the Computer

Computer managed instruction capabilities can include student attendance, student scheduling, student records and student progress. Grade-book record keeping is also available. Many instruction programs include built in management systems. Several major publishers now market sophisticated computer managed instruction systems for education.

In the area of computer-assisted instruction, there are five categories. They include drill and practice, tutorial, simulation, information retrieval, and problem solving. One of the problems that must be addressed is the documentation of educational software. There must be a systematic description of how to use a particular program, how to find the contents of the program on a cassette or disk and recommendations on how this program can be used with a workbook format.

Drill and practice is the most common microcomputer application and for it, there are many commercial programs available. Any novice programmer can easily develop programs of this type.

Tutorial microcomputer instruction is very expensive to develop. Tutorial instruction is carefully developed concept instruction. Graphics and sound are helpful to make meaningful presentations. These programs require an average professional development time of 90 hours for one hour of actual instruction. This area holds much promise for education.

Simulation activities are unlimited. The "Lemonade stand" is a popular example. This activity provides the user an opportunity to run a lemonade stand with the computer providing many decision alternatives. Presently the development is more commercial and game oriented. But, educational possibilities are limitless.

The ability to access information from networks, cable television or telecommunications via satellite also has unlimited potential for education. Cable television is growing very rapidly. There were one million users in 1980, a number which is expected to grow to 12 million by 1981. Video disk and video text capabilities make this an option with a lot of exciting potential.

The last CAI application is problem solving. Remember, problem solving is included as the number one recommendation by NCTM in its agenda for action.

"Problem solving be the focus of school mathematics
in the 1980's"

Problem solving with the microcomputer involves the ability to control the machine. Therefore programming a home computer in its language is essential. The most popular language is BASIC. PASCAL, another language, also has much potential.

The California framework defines problem solving with seven general descriptions. Here are some microcomputer problem solving ideas for each.

1. To develop an understanding of a broad range of strategies.

Problems which require the strategies of guessing, estimating, experimenting, graphing, formulas, etc...can all be explored with the computer.

Example:

$$\text{Formula } A = P(1+i)^n$$

Suppose interest is earned at 6% per year, compounded semi-annually. Find the amount for \$225 deposited for 3 years. Students program the computer to solve the problem. Experiment with different options and estimate the results.

Example:

Compute π as decimal using different formulas

Egypt 1800 B.C. The Rhind Papyrus used $(\frac{4}{3})^3$ for π

Greece 150 A.D. Ptolemy used $\frac{377}{120}$ for π

India 510 A.D. Aryabhata $\frac{62832}{20000}$ for π

Series:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots$$

$$\frac{\pi^2}{6} = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots$$

$$\frac{\pi^2}{8} = 1 + \frac{1}{3^2} + \frac{1}{5^2} + \frac{1}{7^2} + \dots$$

Discuss accuracy of answers as well as different method details.

Other strategies which can be applied with a microcomputer are tables and graphs.

2. To translate words into mathematical symbols.

Students can take word problems and write programs to solve the problem. A major microcomputer task is to learn to program the computer. This is a problem solving experience of communicating with the machine in the language it understands.

Example:

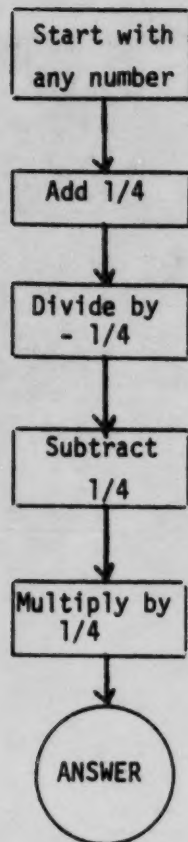
What number am I?

CLUES Divide me by 20
 The quotient is 36
 The remainder is 14

Write a program to solve this problem.

Example:

What is the answer



3. To form mathematical models from real world situations.

Many applications can be placed into this category.

Example:

Find the total bolts of wallpaper needed to cover the inside of a room including the ceiling.

In this case the computer program includes the ability to subtract window and door spaces. Encourage using an input statement so it will solve any wallpaper problem.

4. To develop deductive or convergent thinking skills.

Of course programming a computer involves logical reasoning which is considered an excellent way to develop problem solving abilities.

Example:

Write a program to simulate the sieve of Erathosthenes. This problem involves deductive planning skills.

Example:

Find the sum of the first N numbers. The formula or logical sequence method can be applied using a computer program.

5. To Improve and enhance decision making skills.

Consumer related problems of the best buy or applying conditions to problems can improve these decision making skills.

Example:

Which is the best buy

51¢ 6 oz.	99¢ 1 lb 2 oz	\$2.39 2 lb 8 oz	\$1.89 1 lb 13 oz.	84¢ 1 lb 4 oz
--------------	------------------	---------------------	-----------------------	------------------

Writing a general computer program will help students make decisions on detecting the best buys. A metric equivalent example would also be meaningful.

6. To motivate students to learn and appreciate certain mathematical skills.

Interesting examples can motivate students to solve problems.

Sports, music, cooking, etc...are of interest.

Example:

American League Standings

<u>Eastern Division</u>	<u>Wins</u>	<u>Losses</u>
Baltimore	36	21
Boston	34	21
Milwaukee	32	27
New York	31	27
Cleveland	27	28
Detroit	25	26
Toronto	16	42

Develop a program to print out the standings allowing daily updates.

7. To develop mental and logical skills, related to creative and divergent thinking.

Programming the computer with certain commands lends itself to this ability. Writing programs with loops, IF THEN statements, counter ($I = I + 1$) and sums ($X = X + Y$) make thinking match the "thinking" of the machine rather than general mathematics.

Example:

An ancient king left 17 horses to his three sons. He left $1/2$ of them to his first son. $1/3$ went to the second son and $1/9$ to the third.

However, none of these fractions time 17 gives a whole number of horses. So one young wise son borrowed a horse from a neighbor. He did these calculations:

First Son $1/2 \times 18 = 9$
Second Son $1/3 \times 18 = 6$
Third Son $1/9 \times 18 = 2$

He then returned the eighteenth horse to the neighbor. The three sons walked off happily with 9, 6 and 2 horses.

Write a computer program to show what is wrong with this story.

Other problem solving strategies which lend themselves to computer capabilities are as follows:

Simulations with random capabilities

Tossing coins
Rolling dice
Choosing at random a card

Flow Charting

Prediction by representing situations graphically

Pollution
Smoking
Odds

Looping

Tree growth
Interest accumulations

132

These are some problem solving experiences which the computer can enhance. This educational option is potentially the most significant in terms of problem solving.

In order for these options to successfully be carried out in the classroom, two points are a must: Computer awareness and the ability to program a computer.

Some Final Thoughts:

If a school has a need for a microcomputer, the selection criteria should be primarily on software. Investigate the kind of software that would best benefit the user, and then select the hardware that has that particular software available. Parent groups are a powerful source of funding and a viable booster group for introducing microcomputers into the schools.

Future developments in graphics, sound or speech synthesis capabilities, good library management systems, home information systems and voice recognition will improve the potential of microcomputers in the classroom.

All of these areas have the continuing potential of using microcomputers in the classroom and of improving children's problem solving skills.

Remember!

A computer has no conscience
It feels no passion
It has no life of its own nor any sense of value
It is neither a poet or philosopher
A computer in short is a machine, not a fool or a tyrant.

Source unknown

Bibliography

- Bitter, Gary G., Computer Applications for Calculus, Prindle, Weber
and Schmidt, Boston, 1972, Revision 1982.
- Bitter, Gary G. "Computers and Computer Resources", TEACHER, 97 (5): 62-63
(1980).
- Bitter, Gary G. Computers in Today's World, unpublished manuscript, 1981,
400 pages.
- Bitter, Gary G. Exploring with Computers, Julian Messner, New York, 1981.
- Bitter, Gary G., et.al, McGraw Hill Mathematics Series, Grades K-8, 1981.

STRATEGIES FOR GETTING STARTED IN MICROCOMPUTERS

by
Patrick J. Carr
Tuba City High School
P.O. Box 15E
Tuba City, AZ 86045

This morning I would like to share with you some thoughts and observations that I have been entertaining over the past few years regarding the use and application of technology for educational purposes. While I will be speaking of technology in a general sense I hope that it can be related to the specific concerns brought about by the introduction of microcomputers. The use of technology for educational purposes is nothing new and hopefully we can learn from past mistakes thus avoiding the need to repeat them.

I do not appear before you as someone with all of the answers, to all of the questions, that you probably have on getting started in microcomputers. What I do have though, is some food for thought that I will be serving to you through the frame of reference of a media manager in a medium-sized high school with 1,500 students. I hope that some of this information will be useful to you as school administrators in developing some realistic strategies for obtaining microcomputers to use in various school applications.

The thoughts and experiences that I am sharing with you are not limited to what is happening in my particular situation. Like most schools ours is aware of the potential of microcomputers but not quite ready or able to leap at the first sales pitch delivered our way. We are watching the market and trying to assess what is going on to meet our needs and solve our problems. As many of you know by now, a lot is happening and things are pretty hectic.

As an education specialist for audio-visual services I

have come in contact with various people involved in delivering technology to an educational setting. Salespeople, maintenance personnel, those that service and install school sound systems, intercom systems, public address systems, stage lighting systems, and video systems for closed-circuit television. I have attended many conferences and workshops and spent countless hours with colleagues and other educators discussing mutual concerns for audio-visual services. In addition I have tried to keep abreast of the latest developments through various popular magazines and professional journals. My experiences are not exactly typical.

To begin with, let me say that I consider the microcomputer to be merely another tool in our collection of technology and as such is only as good as the person using it. It does offer us, however, a great deal of promise and potential if we use it wisely.

I want to suggest caution, however, against getting carried away with the notion that somehow computers will cure all that ails us. I do not believe, as some are already proclaiming, that computer literacy will be the next crisis in education or that it needs to become a crisis. If we can keep our heads and blend microcomputers in with our present efforts by considering the computer as another helping tool then there is a good chance that they will be welcomed and accepted and not wind up as another "flash in the pan."

Most technological innovations in education seem to me to be met at first with a flame and end with a fizzle. There is a tendency to get all excited about them at first and then jump to something else when the novelty wears off.

It took almost 400 years since the invention of the printing press until such time as books were used as tools for mass education.

Today most of us are not even remotely aware of the complicated

process involved in filling the pages of our daily "news" papers. Even less are aware of the process of developing and delivering televised or video messages.

Radio and later television to some threatened the very existence of schools, colleges, and universities when they were first introduced. In their places were envisioned the "University of the Air" and its related versions. Those who have been watching developments far closer than I tell me that in spite of a thirty-five year history, television and radio have failed miserably to live up to the promise and potential that accompanied their introduction.

As I look over the history of education and technology there appears to me to emerge a phenomenon that I have called an "application gap". It exists between the introduction of an item and its widespread acceptance and application. I call it to your attention not to put a damper on progress but as a warning that things can be approached with a cool head.

For long we have known the promise and potential of computers in education. Up until a few years ago computer technology was too expensive and remote for widespread application. Most schools simply could not afford computers in their programs. Today, however, as the technology increases and the costs decrease most schools are starting to realize that they cannot afford not to have them in their programs.

How do we get started? That is what we are all here for are we not? Let me offer some suggestions.

First of all you need to determine if you really can afford to get started with microcomputers at this time. Will your present program resources, in terms of manpower and money, allow for them?

Costs are deceptive at times. Some schools have already laid out big bucks for a computer system only to find out that they could not afford the necessary peripherals and software to run them.

What sort of applications will you be making with the microcomputers in your school? Will they be used to teach "computer literacy" or for data processing, electronic filing, word processing, financial reporting, computer assisted instruction, computer managed instruction, or accessing large data banks.

What are your immediate needs? Will you want microcomputers for administrative purposes? ie. attendance, grading, payroll or scheduling.....

Will you want them for academic purposes? Computer Assisted Instruction or Computer Managed Instruction?

Will they be used primarily in the Math/Science Dept. or in Business? Language Arts, Fine Arts, or Graphic Arts? Will they be used to teach "computer literacy" or computer construction? Will they be used to challenge the gifted or encourage the slower students? You need to know because all microcomputers are not the same and the particular application should determine the type that you obtain. Wise planning will prevent them from winding-up in a corner, gathering dust-a fate of many technological innovations in the past.

To assist in wise purchasing some schools have selected one person to oversee or coordinate computer purchases so as to avoid duplication of effort and incompatible equipment purchases. Conventional suppliers of data processing services for the school or district financial and administrative needs have a tendency to be hung-up on their own systems and feel threatened that the

microcomputers will somehow invade their domain.

Department chairpersons tend to respond to their particular needs and cannot objectively appreciate the needs of other departments.

Traditionally trained, book-orientated librarians may be in need of additional up-dated training if you can get them interested in microcomputers.

Audio-visual departments or school media centers are often the logical place to coordinate computer needs but personnel will need training to update their skills.

Whoever purchases the equipment, however, needs to involve the potential users and to arrange for hands-on demonstrations and training for users. Computers should not be selected on the basis of their pretty pictures in a fancy catalog or brochure.

I want to mention the importance of compatability of items and software because it is important. We had all sorts of problems with it when using video equipment until the manufacturers got their act together and started to standardize pieces for the customer's convenience. It is probably too early to expect much in this area by the computer manufacturers for in any period of rapidly expanding technology this is the furthest thing from their game plan.

Let's face it. Most educators are not personally or professionally prepared to make some of the complex decisions surrounding the purchasing and designing of microcomputer systems and programs. Alexander Pope had us in mind, for sure, when he said that a little knowledge is a dangerous thing.

Also, we do not have the time to keep up to date on the latest trends and developments or to write complex programs to meet our needs. The need for communication and coordination with others is obvious.

In the past many inquiring educators have been turned-off and discouraged in their dealings with the technological wizards. Too often you ask for the time and they tell you how to build a watch. Do we really need a lecture on microcomputer circuitry when we ask how one works? We can proudly acknowledge our limitations without coming off as some sort of a jerk.

I encourage all school people to keep abreast of the latest trends and developments so that they will know which tools are available to help them. At the same time I encourage those providing the services to realize that not everyone is into their particular specialty. If there is a communication problem they have as much to lose as the person who doesn't understand what they are talking about.

Let me now discuss some of the outsiders who we might call upon for help.

Salespeople. While not wishing to question the integrity or intentions of any individual salesperson, I think that it is fair to say that the bottom line, as far as these people are concerned, is how much money they can make. This is not necessarily a bad thing but it is after all the nature of the beast and we need to recognize it as such.

Too often we educators insist on paying rock-bottom dollar for equipment and then wonder why we are not able to get the maintenance, training, and service needed at a later date.

Also, we need to keep in mind that salespeople have a tendency to sell items that we might not necessarily need or that will require

additional purchases in the future. It is always advisable to get a second opinion, preferably from a competitor, whenever making a major purchase. You will need to struggle with the differences should they emerge. Life is tough!

Also, try to purchase equipment that will meet your unique needs and not those of some other school across town. Too often salespeople have little understanding of school needs. All schools are not the same.

In the course of my dealings with salespeople I meet those who are pushing video equipment one week and fishing tackle the next. They have a big broad smile on their faces as they line up to market the latest in the computer line.

In addition and in all fairness, there are those who are genuinely interested in helping and have been doing so for many years now. They are not hard to find, seek them out, patronize them and tell us all about the good services that they perform.

Colleges of Education. Here is the logical place to turn to for help. Some, however, have been distracted and misled by too many gadgets and gimmicks in the past to get too excited about another fad. As I looked over the program lined up for us during the next two days it is rather obvious that Arizona State University is not one of those. These institutions need to take the initiative and the lead in developing adequate coursework to meet the needs of teachers in the schools.

College Students. Nice guys. Willing to help. Not too expensive, but... Some school administrators have explored this area with varying degrees of success. Computer science students can help but you need to get documentation so that when they leave the program doesn't leave in their heads.

School Staff. Some schools have been turning to their staff members personally interested in the new technology as a means of getting started. While they are not strictly outsiders they have an interest in something beyond what is normally expected of them and can be utilized as such. Most successful innovations usually have as ingredients of their program involvement of the staff that will ultimately make use of their efforts. What use are your best laid plans if you cannot get your staff to implement them?

Also, in getting started in microcomputers you will need to work in some inhouse training for your staff. Most of your teachers come from an age known as BC (before computers) and will probably have some fears and misconceptions about microcomputers. They need to know that you don't need to produce a movie to use a projector nor do you need to program in order to use a computer. Some sort of orientation to the new technology is practically a must. Many computers today operate on the "read-and-run" method so all you might need to do as an administrator is to set one up in a lab somewhere and encourage your teachers to use it.

Computer Consultants. If you can find consultants who have not been gobbled up by business or industry hang on to them. They can be very helpful though they might not have much of a track record with microcomputers because of their recency. You might have to lay out a few bucks for their services in the form of a systems analysis of your situation, but it is money well spent. Some things that you might want to keep in mind in dealing with these fellows are:

1. Do not hire them by the hour. It is best to have them

help you identify your particular needs and work with you in developing a total package. Be sure that you have all of the commitments in writing should you have any concerns at the end of the project. Also, if they write any programs for you be sure that they are well documented so that you will not be locked-into a program that cannot be changed.

2. Avoid consultants who promise to do for you what they did for someone on the other side of town. Your needs are not going to be the same though there might be some similarities.

3. Check-out your consultant with others who have used his services and if you can afford it try to get opinions from other consultants as to how best solve your computer needs.

Computer Journals. There are several microcomputer journals on the market today that should become a part of your subscription list. They can be placed in the computer lab or in another place accessible to the students and staff. Let me mention a few that appear to have some future:

BYTE Magazine

Interface Age

Creative Computing

Personal Computing

Educational Computing

In addition many local communities are starting their own microcomputer clubs and publishing some worthwhile newsletters. You can learn all about them through your local computer stores.

PROFESSIONAL ASSOCIATIONS. Some professional organizations are emerging with programs to provide leadership and training for the schools. Among these would be:

Computer Using Educators

Association for Educational Data Systems

Association for Education Communication & Technology

As an administrator you need to realize the impact that microcomputing technology will have in your school and on your programs. It is an exciting and challenging time for education not a time to get discouraged. We will need a lot of help and coordination with many people outside of our school situations. When you ask them for the time don't let them tell you how to build a watch.

SOME FOOD FOR THOUGHT...

"THE APPLICATION GAP"... to what extent does the technology precede widespread application... ?

BEFORE PURCHASING A MICROCOMPUTER YOU MIGHT WANT TO CONSIDER THE MAIN PURPOSE FOR WHICH IT WILL BE PUT TO USE:

- ADMINISTRATIVE PURPOSES? -COMPUTER ASSISTED INSTRUCTION? -COMPUTER MANAGED INSTRUCTION? -PRIMARILY IN THE MATH/SCIENCE DEPARTMENT?
- IN THE BUSINESS DEPARTMENT? -IN LANGUAGE ARTS? -FINE ARTS?
- SPECIAL EDUCATION? -TEACHING COMPUTER LITERACY? CONSTRUCTING COMPUTERS? ETC.

The particular purpose or application will determine the computer system that you will want to purchase, so plan wisely. One system will not serve all purposes.

COMPATIBILITY OF COMPONENTS WILL ENABLE YOU TO STRETCH YOUR COMPUTER DOLLARS.

MOST OF US MORTALS IN EDUCATION ARE NOT PERSONALLY OR PROFESSIONALLY EQUIPPED TO MAKE COMPLICATED TECHNOLOGICAL DECISIONS. MOST OF THE TECHNICAL WIZARDS ARE NOT EQUIPPED TO MAKE SOME OF THE DECISIONS THAT WE AS EDUCATORS HAVE TO MAKE. WE NEED TO DEPEND ON OUTSIDERS TO HELP US DESIGN AND DEVELOP USEFUL COMPUTER SYSTEMS AND PROGRAMS.

SOME SOURCES OF HELP MIGHT INCLUDE:

- | | |
|--|---|
| -COLLEGE STUDENTS
(in computer sciences) | -COLLEGES OF EDUCATION
(if only they were all like ASU) |
| -HARDWARE SALESPERSONS
(tenga mucho cuidado,
amigos) | -INTERESTED STAFF MEMBERS
(may have more to offer than you
think and are often important
ingredients of successful innovation) |
| -COMPUTER CONSULTANT FIRMS
(pay them well so that they are
not driven off into business or industry.
and get guarantees on their work.) | |

... keep abreast of the latest developments and mindful of your limitations so that in dealing with outsiders they will not be telling you how to build a watch when you are merely asking for the time...

A MICROGLOSSARY ON MICROCOMPUTING

The following are a few "buzz words" that will help you get started in coming to terms with microcomputers. The list should be expanded as you become more familiar with the technology.

ACCESSORIES- see peripherals

ALPHANUMERIC INFORMATION- numbers and letters

BASIC- Beginners Allpurpose Symbolic Information Code. One of several microcomputer languages.

BUG- errors or mistakes.

CAI- Computer Assisted Instruction.

CMI- Computer Managed Instruction.

CAL- Computer Assisted Learning

CBE- Computer Based Education

CPU- Central Processing Unit "the brain".

CHARACTER- any individual, letter, number, or symbol.

COMPUTER LITERACY- skills needed to understand computers.

FLOPPY DISK- Shaped like a small 45 rpm record-disk. They are in different sizes. Store computer programs for speedy access.

HARDWARE- the equipment.

INPUT/OUTPUT- processed and unprocessed data

INTERFACE- link-up or connections with other hardware

MEMORY- storage

MODEM- modulator/demodulator allows computers to interface other computers and data banks via phone lines.

K- prefix "kilo" means 1,000. Actually K equals 1,024 bits.

PERIPHERALS- accessories

RAM- random access memory... main memory of computer that must be stored or will be lost when computer is turned off.

ROM- permanent program that tells computer how to function

SOFTWARE- computer program

SIMULATIONS- band of options with logical sequences composed of circumstances, decision, results, new circumstances, that can be controlled by a computer.

TERMINAL- keyboard and screen usually remotely connected to computer.

MICROCOMPUTING PURCHASING PLAN

1. Can you afford computerization at this time? (Resources: money and manpower?)
2. Will microcomputerization really help you achieve stated goals?
3. What sort of computer will we purchase to do the job?

POPULAR COMPUTERS

BASIC PACKAGE COST PERIPHERALS
AVAILABLE SOFTWARE MAINTENANCE DOCUMENTATION ETC.

APPLE II & III

ATARI

TRS-80 & Models

TEXAS INSTRUMENTS

Bell & Howell
(Apple)

Scott, Foresman & Co
(Texas Instruments)

The above chart is merely suggested as a strategy that might be employed for a more informed choice on purchasing microcomputers. Other items of concern can be added or substituted across the top and other computers can be listed or substituted in that column. No attempt was made to add information into the boxes as that information is rapidly changing and may not be accurate at the time of your purchase.

III. COMPUTER ASSISTED INSTRUCTION

D. V. HIGH COMPUTERS

Instructor: Jeff Martin
Business Dept: Rich Russell
Principal: Dr. L.A. Kope

Dear Valley High School
18424 N. 51st Ave.
Glendale, AZ 85308
(602) 995-7475

COMPUTER CURRICULUM

Course Title: Introduction to Computers
Department: Business Education
Length : Semester - 1 hour
Offered : '80 - '81

Description - Hardware, software, flowcharting, design, programming, data processing, word processing, process control, crime, social impact, and careers in computers are all covered lightly.

Course Title: Computer Skills
Department: Business Education
Length : Semester - 1 hour - 9 weeks Typing, 9 weeks electronics
Offered : '81 - '82

Description - This course is designed to a) provide students with skills needed to efficiently interact with a computer keyboard, b) give students an introductory taste of technology and maintenance work, and c) keep students informed of current events in the computer field.

Course Title: BASIC Programming - Business & Math
Department: Bus. Education & Math Education
Length : 1 semester - 1 hour
Offered : '81 - '82

Description - Fundamentals of programming in the BASIC language. Algorithms, current events, and problem solving will have either a Business/Industrial or Math/Science emphasis.

Course Title: Advanced Programming
Department: Math Education
Length : 1 semester - 1 hour
Offered : '81 - '82

Description - More BASIC and an introduction to PASCAL. A substantial project the 2nd nine weeks allows students to chose between more PASCAL programming, Assembly language programming, and building computers from kits.

Course Title: I.C.E. - Industrial Cooperative Education
Department: Industrial Arts
Length : 1 year - 2 hour block morning - afternoon work experience
Offered : '82 - '83

Description - Computer and business topics covered in morning, afternoon work.

MICROCOMPUTERS IN BUSINESS EDUCATION

WILLIAM DRUM, DEPARTMENT CHAIRMAN
BUSINESS EDUCATION
RINCON HIGH SCHOOL
422 N ARCADIA BLVD.
TUCSON AZ 85711

"BUSINESS EDUCATION MUST DO MORE TRAINING IN DATA PROCESSING." STUDENTS MUST BE TAUGHT BUSINESS APPLICATIONS ON MICROCOMPUTERS." THESE STATEMENTS ARE EMPHASIZED OVER AND OVER IN THE CURRENT LITERATURE AND GATHERINGS OF BUSINESS TEACHERS AND I HEARTILY CONCUR. FOR TOO LONG THE PRIMARY EMPHASIS HAS BEEN MATH APPLICATIONS AND GAMES.

SINCE 80 PERCENT OF THE WORK DONE ON COMPUTERS IS IN THE FIELD OF BUSINESS, DATA PROCESSING HAS HAD A STRONG EMPHASIS IN THE BUSINESS CURRICULUM AT RINCON FOR SEVERAL YEARS. THIS IS NOT TRUE IN MANY SCHOOLS, HOWEVER. I FEEL THERE ARE SEVERAL REASONS FOR THIS: (1) COST OF THE EQUIPMENT (2) TEACHERS DO NOT HAVE THE NECESSARY BACKGROUND OR TRAINING (3) LACK OF BUSINESS ORIENTED MATERIALS. WE HAVE BEEN FORTUNATE ALSO FOR HAVING THE SUPPORT OF THE ADMINISTRATION.

WITH THE LOW COST OF THE MICROCOMPUTERS, THE FIRST OBSTACLE IS OVERCOME. I RECOMMEND AT LEAST A 32K DISK SYSTEM WITH PRINTER. WE HAVE THE TRS-80 MODEL I, LEVEL II WITH TWO DISK DRIVES AND PRINTER. WE ARE PLANNING TO ADD THE NEW MODEL III.

AS FOR THE LACK OF TRAINING, THERE ARE MANY WORKSHOPS NOW BEING OFFERED BY THE UNIVERSITIES AND EVEN THE COMPUTER STORES. I WILL BE CONDUCTING ONE AGAIN THIS SUMMER AT THE UNIVERSITY OF ARIZONA FROM JUNE 22 TO JULY 10 SPECIFICALLY FOR BUSINESS TEACHERS.

THE PUBLISHING COMPANIES HAVE RECOGNIZED THE NEED FOR MATERIALS WITH BUSINESS APPLICATIONS AND SOON TEXTBOOKS AND OTHER MATERIALS WILL BE ON THE MARKET. IN THE MEANTIME WE NEED TO DEVELOP OUR OWN.

MICROCOMPUTERS, PRODUCTS OF THE LAST FEW YEARS, PROBABLY WILL HAVE A GREATER IMPACT ON THE OFFICE AND THE SMALL BUSINESS THAN ANY OTHER DEVELOPMENT. THEY ARE PUTTING TREMENDOUS COMPUTING POWER WITHIN THE REACH OF ANY BUDGET. SYSTEMS WITH CRT'S ARE AVAILABLE FOR LESS THAN \$1000 AND SMALL SYSTEMS COMPLETE WITH DISKETTE STORAGE AND HARD COPY PRINTERS CAN BE OBTAINED FOR \$3000 AND UP. PACKAGED SOFTWARE OF TYPICAL BUSINESS APPLICATIONS--GENERAL LEDGER, PAYROLL, INVENTORY, ACCOUNTS PAYABLE AND RECEIVABLE, ETC--ARE AVAILABLE FOR A NOMINAL COST. WORD PROCESSING SOFTWARE IS ALSO AVAILABLE FOR THESE SYSTEMS THUS ALLOWING THEM TO SERVE A DUAL FUNCTION IN BUSINESS EDUCATION.

MOST MICROCOMPUTERS ARE PROGRAMMED IN BASIC. HOWEVER, SOME MANUFACTURERS ARE NOW DEVELOPING AND MARKETING COMPILERS IN

FORTRAN AND VERSIONS OF COBOL. THIS NECESSITATES ADDING MORE MEMORY BUT MOST MICROCOMPUTERS CAN BE EASILY EXPANDED.

MICROCOMPUTERS ARE GAINING A FOOHOLD IN LARGE BUSINESSES, TOO. SOME OF THE PROBLEMS OF HAVING A CENTRALIZED COMPUTER SYSTEM, AS WELL AS THE COSTS OF COMMUNICATION, HAVE BEEN REDUCED BY DECENTRALIZING AND USING MICROCOMPUTERS.

NOT ALL STUDENTS WILL BE PROGRAMMERS, BUT ALL STUDNETS WILL BE WORKING WITH THE INPUT AND OUTPUT FROM COMPUTERS; AND SOME OF THEM WILL COME IN DIRECT CONTACT WITH INPUT AND OUTPUT DEVICES. THEY MUST BE ABLE TO USE THEM. MANY SECRETARIAL AND CLREICAL WORKERS WILL BE USING TERMINALS TO SET INFORMATION FORM THE COMPUTER AND TO ENTER DATA INTO IT. THEREFORE, THESE PEOPLE NEED SOME TRAINING ON THE USE OF TERMINALS (OR MICROCOMPUTERS). SMALL UNITS OF INSTRUCTION CAN BE INCORPORATED IN EXISTING SECRETARIAL AND CLERICAL TRAINING CLASSES. TODAY'S ACCOUNTANT IS USING THE COMPUTER AS A TOOL. THERFORE, ACCOUNTING COURSES MUST INCLUDE UNITS OF INSTRUCTION UTILIZING THE COMPUTER. AGAIN, THESE STUDENTS DO NOT NEED TO BE TAUGHT PROGRAMMING BUT MUST LEARN TO USE ACCOUNTING APPLICATIONS.

ALL BUSINESS STUDENTS SHOULD TAKE AT LEAST AN INTRODUCTION TO DATA PROCESSING. MANY TEXTS AND MATERIALS ARE AVAILABLE FOR A COURSE OF THIS NATURE.

THE TOPICS COVERED IN A GENERAL INTRODUCTION SHOULD
INCLUDE:

- A. AN INTRODUCTION TO THE COMPUTER AND CAREER
POSSIBILITIES
- B. THE EVOLUTION OF THE ELECTRONIC COMPUTER INDUSTRY
- C. PROCESSING DATA ON A COMPUTER SYSTEM
- D. THE PROCESSOR UNIT
- E. INPUT MEDIA--ADVANTAGES AND DISADVANTAGES OF
VARIOUS TYPES
- F. OBTAINING OUTPUT
- G. AUXILIARY STORAGE AND FILE ORGANIZATION
- H. DATA COMMUNICATIONS
- I. DATA BASE AND DISTRIBUTED DATA PROCESSING
- J. SYSTEMS ANALYSIS AND DESIGN
- K. PROGRAM DESIGN AND FLOWCHARTING
- L. PROGRAMMING LANGUAGES
- M. THE FUTURE OF COMPUTERS IN SOCIETY

MUCH SOFTWARE IS AVAILABLE FOR TYPICAL BUSINESS
APPLICATIONS. THE CHART ON THE NEXT PAGE SUGGESTS SOME
SOFTWARE THAT IS AVAILABLE, IN WHAT CLASSES IT CAN BE USED
AND SOME OF THE ACTIVITIES. THE SOFTWARE PACKAGE COMES
WITH THE PROGRAMS AND THE INSTRUCTIONS FOR USING THEM. THE
INSTRUCTOR MUST PREPARE SOME ACTIVITIES TO ACCOMPANY THIS
SOFTWARE AND GIVE THE STUDENTS PRACTICE IN USING IT.

APPLICATIONS USING SOFTWARE ON THE MICROCOMPUTER

Application	Task	Performance
Mailing List Software	Given a list of names, addresses and select codes	the student will create and store on disk a mailing list.
Use in:		
Clerical Practice	Given an existing mailing list stored on disk and a list of changes, additions and deletions	the student will update the mailing list, sort it in alphabetical order and prepare a listing.
Secretarial Practice		
Office Practice		
Data Processing	Given blank labels (or paper) and an existing mailing list	the student will print mailing labels for all addresses with a selected zip code (or any other select code).
Word Processing Software	Given a short rough draft document	the student will create a file, revise it, make additions to it, and have it printed with the margins justified.
Use in:		
Clerical Practice		
Secretarial Practice	Given a two-page handwritten report with revisions	the student will produce a final printed document with the margins justified.
Office Practice		
Data Processing	Given a rough draft form letter	the student will produce a mailing.
General Ledger Software	Given a setup for a service type business and a list of transactions for a month	the student will record the transactions and prepare the income statement and balance sheet for the fiscal period.
Use in:		
Accounting	Given a setup for a merchandising type of business and a list of transactions for a month	the student will record the transactions and prepare the income statement and balance sheet for the fiscal period.
Accounts Receivable Software	Given a list of names and addresses of customers	the student will create an accounts receivable file.
Use in:		
Accounting	Given a list of transactions, changes, additions and deletions	the student will update the accounts receivable file and prepare monthly statements and a schedule of accounts receivable.

Most of the programming on the microcomputers is done in BASIC, if the student is going to be taught to program, then business applications should be used. The following programming topics should be covered:

1. Getting started in BASIC
 - a. introduction to keywords
 - b. variable names
 - c. mathematical operators
 - d. order of operations
2. Doing more with BASIC
 - a. introduction to flowcharts
 - b. headings
 - c. READ and DATA statements
 - d. loops and use of IF statement to check for trailer
3. Improving the appearance of reports
 - a. use of TAB statement
 - b. formatting with PRINT USING statement
 - c. string variables
 - d. planning a layout on the video display sheet
4. Accumulating totals and comparing
 - a. single and multiple totals
 - b. relational operators
 - c. comparing, decisions and branching
5. Responding to the computer
 - a. INPUT statements
 - b. interactive programming
6. Creating loops
 - a. FOR/NEXT statement
 - b. creating charts
7. Taking subtotals
 - a. control breaks
 - b. minor and intermediate totals
 - c. subroutines
8. Storing Data in tables
 - a. arrays
 - b. sorting
9. Creating, maintaining and using data files
 - a. storing data on disk
 - b. accessing stored data
 - c. updating data files
10. Using graphics (optional)
 - a. boxing reports
 - b. bar graphs

Today's office is rapidly changing, and we as business

teachers must be ready and willing to keep pace with the times. It may mean additional training for ourselves, it may mean a drastic revision of our curriculum, it may mean a "sales Job" on the administration convincing them of the need for new equipment, and it may take lots of hard work; but it is challenging, exciting and rewarding.

A Middle School Microcomputer Language Arts Series

ENGLISH USAGE EXERCISES

Robert L. Williams, Ed.D.
St. John's University
Jamaica, New York 11439

English Usage Exercises is a CAI series being developed by John G. Allee, Jr., The George Washington University, Peter J. Quinn, St. John's University, and Robert L. Williams, St. John's. The series concerns itself with written usage, although elements of speech syntax are frequently called upon to help demonstrate written usage practices. The series follows a "spiral-ing" design: topics, after their first presentation, are visited again after other topics are treated, each visitation presenting a new or additional concept. Most topics are presented by demonstration and example; drill and practice follow. Throughout the series, an especially heavy emphasis is given to demonstrations of "grammar at work." Although grammatical systems underlie the presentation of usage, few grammatical terms are thrust upon the student: instruction is by example and observation, not by definition. While the series is intended for grades 4 through 6, it should also prove useful for higher grades and levels, especially where "reluctant" students may need remedial work in basic language skills.

Two volumes of English Usage Exercises are ready for use in the upper elementary and junior high grades. These first two volumes primarily explore subject-predicate relationships that often reflect written usage problems for many students. The first volume gives major attention to predicates--verb constructions. The second volume gives major attention to subjects--nouns and pronouns. Most sessions ask the student to supply verb, noun, or pronoun forms to "fill" blank slots of the exercise sentences, although some require descriptive responses like "past" or numeric responses like "1." Nontraditional terms like "counting" and "keeping time" are used to help keep language discussions concrete.

The role of language arts Computer Assisted Instruction is not clearly established. It seems to us that it is appropriate to consider CAI simply another tool of study--in a sense, a dynamic textbook. Clearly CAI should be a tool of teachers as well as of, and for, students. And computer resources dedicated to CAI might well be considered extensions of traditional school library resources. We believe that many teachers can use CAI effectively with their students, if they choose to do so, both to their own advantage and to that of their students. CAI allows the student to study dynamically with privacy, a factor sometimes not adequately assessed. Though corrected by the

computer, the student is not publicly exposed. The student is treated with consistency by the computer: for some students, particularly those who have been reluctant learners in the past or who have had problems relating to teachers, the computer is invitingly neutral and unemotional. The student is simply assessed and rewarded without bias. The computer sees no skin color nor does it hear any speech dialect: the student and computer communicate and relate without prejudice, through uninflected alph-numeric characters, words, or numbers. Thus, CAI is decisive but unemotional, instant in reply but patient in turn, an equal-opportunity tutor.

CAI ought to assist the teacher, and English Usage Exercises does. Both volumes of English Usage Exercises are supported by guides that display all of the sessions. These guides also discuss the content and organization of the sessions. In effect, the guides provide the teacher with easy to handle, comprehensive summaries of the work to be met by the student. Also, rather importantly, the guides can be used as "old-fashioned" lesson plans by the teacher for classroom sessions, if need be. Outlines of

the lessons of English Usage Exercises follow:

LESSONS OF VOLUME I

<u>Series</u>	<u>Lesson Titles</u>	<u>Number of Lessons</u>
1	Counting with Pronouns	9
2	Counting Throughout Sentences	2
3	Double Subjects	4
4	Keeping Time	7
5	'Ing' Verbs	6
6	Object Pronouns	5
7	Questions	4
8	S & O Endings	7
9	D Endings or Markers	4
10	Verbs and Questions	3
11	Counting Review	5
12	This & That	3
13	Possession	5
14	It & They	3
15	Subject and Object Pronouns	6
16	Have	3
17	Have & Do	5
18	Helpers	3
19	Contractions--Not	4
20	Contractions--Verb	3
21	'D' Forms	5

LESSONS OF VOLUME II

<u>Series</u>	<u>Lesson Titles</u>	<u>Number of Lessons</u>
22	N Forms of Verbs	16
23	DN Forms of Verbs	12
24	Spelling Regular Verbs	10
25	Questions	10
26	And & Or	8
27	A & The	6
28	Connectors	4
29	O & S Markers	9
30	Countable Nouns	5
31	Pronouns & Connectors	11
32	Possessives	11

Some series have more lessons than others. No attempt has been made to balance the series in length. Rather obviously the contents of the topics determine their needs. In some instances a topic may be a review or may serve as a bridge between two more important topics.

The first 'edition' of English Usage Exercises has been prepared for Model I TRS-80* Level II BASIC Dual or Multiple Disk Systems. This first edition is composed of five disks, disk 1 through 3 housing Volume I, disks 4 and 5 housing Volume II. Each disk contains a directory program that permits the user to select the lesson he desires from that disk. Upon selection, the directory program brings the lesson into core. Until the lesson is concluded or terminated no other disk access is made. That is, each session is a stand-alone program that generates its own exercises. When the lesson is completed, the directory program is returned to so that the student may choose another lesson should he wish to do so.

A demonstration disk is available. This disk allows the interested educator to sample English Usage Exercises. The disk contains a selection of lessons from Volumes I and II and two lessons from Volume III, which is now being authored.

The 100 lessons of Volume III are to be released in September of 1981. The fourth and concluding volume of 100 lessons is scheduled to be released in 1982.

English Usage Exercises is available through 3 R Software. The prospective user may find a copy of one of the two teacher's guides a helpful and economic way to preview the series. Again, to assist in making a decision, the modestly priced demonstration disk provides a unique and efficient way to sample the series. Guides, demonstration disks, and the series are available by writing to:

3 R Software
P.O. Box 3115
Jamaica, N.Y. 11431

COMPUTER LITERACY

Ted Bell
Prescott Public Schools
146 S. Granite
Prescott Az 86301

Course Description

Computer Science for the Gifted

This course is designed to teach the fifted students the fundamentals of writing programs of the micro computer in BASIC language. The written program will be used in the areas of General Business, English, Science, and Math to teach the students application of computers in these areas. Prerequisite: Recommendation of Instructor.

Introduction to Computer Science

The theme of this course will be to explore the general areas of computer science. The emphasis will be in BASIC programming, theory, and careers. Class time will be divided into the following major areas: 65% laboratory, 30% theory, and 5% field trips and appropriate audio-visual enrichment. It is recommended that the students have at least one year of algebra.

1st & 2nd Week

I. Introduction

1. Pre-Test
2. Machine use & care
3. Write programs using read, if-then, for-next etc.
4. Create new programs

3rd-6th Week

II. Write Programs in the Business Areas using the gifted students

7th-8th Week

III. Take the Programs created by the gifted students to teach students in the business area.

1. Introduction in use of computer
2. Instruction on use of the program

9th-11 Week

IV. Write Programs in the Math areas using the gifted students

12th-13th Week

V. Take the Programs created by the gifted students to teach students in the Math area.

1. Introduction in use of the computer
2. Instruction on use of the programs

14th-16th Week

VI. Write Programs in the Science areas using the gifted students

17th-18th Week

VII. Take the Programs created by the gifted students to teach students in the Science area.

1. Introduction in use of computer
2. Instruction on use of the program

VIII. Final rap-up

1. Post Test
2. Discussion and suggestions on next years programs.

INTRODUCTION TO COMPUTERS

High School

Instructor: Fred Bell

Text: Hands on Basic With a PET
by Herbert D. Peckman

1st Week: The PET Computer and BASIC

- a. Pre-test
- b. What is BASIC?
- c. How to Begin
- d. Intro Demo Tape
- e. First lab ($V = L * W * H$)

2nd Week: Using the CRT

- a. Calculator Mode
- b. Screen Editing
- c. Screen Graphics
- d. Cursor Demo Tape
- e. LET Tape and INPUT Tape
- f. Debug Assign. (Mileage to Phoenix)
- g. Lab ($A^2 = B^2 + C^2$)
- h. Computer Costs
- i. Test

3rd Week: Building Programs

- a. Elephant Flow Chart
- b. LOAD and SAVE a Program
- c. Order of Operations
- d. E Notation Worksheet
- e. READ and DATA Statements
- f. READ and DATA Tape
- g. Lab (Gross Pay and Potato Strings)
- h. The Rip Off
- i. Test

5th and 6th Week: Branching

- a. IF THEN Worksheet
- b. GOTO Worksheet
- c. Problems using IF THEN and GOTO
- d. Film
- e. Lab (Centigrade - Fahrenheit)
- f. Test
- g. Careers

INTRODUCTION TO COMPUTERS

CONTINUED

7th and 8th Week: Looping and Functions

- a. Name Printing Using FOR NEXT and GOTO
- b. Lab (Golf)
- c. Assignment Using Function
- d. Other Hardware
- e. Test

9th Week: Creating Random Games

- a. Random Number Worksheet
- b. GO SUB Worksheet
- c. Lab (Random - Multiplication)
- d. Computers in Society
- e. Test

10th and 11th Week: Matrix and Array's

- a. Array's Worksheet
- b. Array Problem
- c. Computers in Society
- d. Resource Person at Yavapai College
- e. Test

12th Week: GO SUBS

13th, 14th, and 15th Week: Creative Projects in Science and Math

16th, 17th, and 18th Week: Creative Projects in Business

POST-TEST

Prestin Bird
Gifted Child 7 Project
2757 W. Pensacola Street
Tallahassee, Fla 32304

Evaluation Summary for the 1979-80 Delivery of
Computeronics

The nonequivalent control group design was used to evaluate the Computeronics course on fifth through eighth grade high achieving and gifted students. Pre and posttests were administered to the experimental and control groups. The Computeronics Criterion Referenced Test (CCRT) with KR-20 reliability of .90 (N=898) was used to assess student knowledge of computers and BASIC programming.

The data was analyzed with an analysis of covariance which statistically adjusts for the difference in the experimental and control group pretest means. Figure 1 displays the increase in pre to post CCRT scores for the experimental group. Table 1 presents the statistics associated with the analysis of covariance for the CCRT. The experimental group showed a statistically significant difference in posttest scores over the control group.

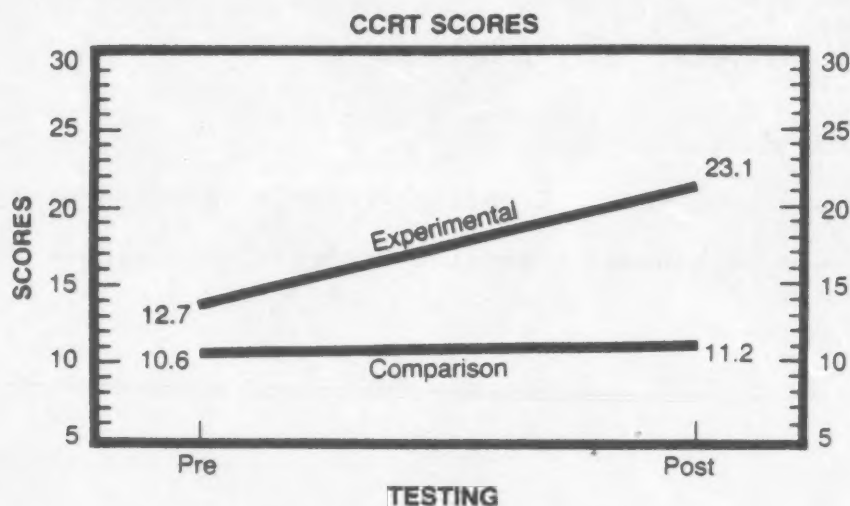


Figure 1. Mean Computeronics Criterion Referenced Test scores for the experimental and comparison group.

Table 1

Pre and Posttest Scores for the Experimental Group and Control Group on the Computeronics Criterion Referenced Test

Group	N	Pretest	Posttest	SD
		$\bar{X}$	$\bar{X}$	
Experimental	658	12.7	23.1	4.3
Comparison	115	10.6	11.2	3.2

Note: The analysis of covariance using this data was statistically significant, $P < .001$.

Gifted Child Project

SO, YOU WANT TO ORDER A SAMPLE?



*Teacher Guide

Activity Packet

Game from any/each module, e.g.:

Wild Duck
Assumption Junction
Convoy/There's A Leak

<u>Module</u>	<u>Price</u>
	\$
Mind Bender	9.55
Critical Thinker	7.05
Strategies	7.90
Inquiry	7.70
Mind Bender	2.40
Critical Thinker	1.50
Strategies	1.40
Inquiry	.65
Mind Bender	4.00
Critical Thinker	8.25
Strategies/Inquiry	4.50

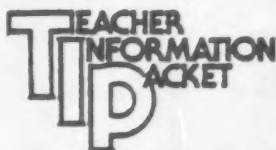
COMPUTERGNICS

*Teacher Guide

Activity Packet

Sidetrips

<u>Module</u>	<u>Price</u>
	\$
Problem Solving With Computers	10.10
Computers in Society	6.60
Problem Solving With Computers	2.10
Computers in Society	.85
Problem Solving With Computers	2.85



Handbook
DOITTOIT Workbook
Perspectives Game

<u>Price</u>
\$
3.35
5.30
3.75

* Teacher Guides include Student Lesson Books and Mastery Answer Books

2.21

GIFTED EDUCATION - - ENRICHMENT OR ACCELERATION? - - COMPUTERS PROVIDE BOTH

Marilyn Sue Ford
Las Vegas Public Schools
Las Vegas, Nevada

Patricia A. Walkington
Gilbert Public Schools
Gilbert, Arizona

Gary G. Bitter
Arizona State University
Tempe, Arizona 85281

The question of acceleration (rate of learning) versus enrichment (development of higher level thinking skills) remains an issue among educators who work with students identified as gifted. But, in order to provide a differentiated educational program for gifted students, a combination of both acceleration and enrichment deserves consideration. The authors suggest a computer curriculum including literacy and programming skills as an example of a program which can be implemented with this dual approach.

The enrichment/acceleration computer curriculum approach was used with fifth through eighth children during the Summers of 1979 and 1980. The program was sponsored by the Tempe (Arizona) Parents Gifted Association (TPG) under the auspices of Dr. Gary Bitter, Professor of Education at Arizona State University, Tempe, Arizona. The program took place at a local high school where students used six typewriter-like terminals connected by telephone line to a large master computer, the PDP 11/70, located at the University. In addition, a local vendor provided a microcomputer for classroom use.

The children learned to program in the BASIC language using BASIC For Beginners, McGraw-Hill, 1978. Several computer literacy topics were introduced during the five-week summer session. The topics included an introduction to the computer, what a computer is and how it works, a history of the computer, different types of computers, the input and output devices of a computer, developing a computer flowchart, and the advantages and limitations of computers in our society.

The students in the program were introduced to the various programming statements and commands through simple, isolated sample programs. As individual children

Enrichment or Acceleration?

mastered the BASIC programming language, acceleration was achieved through sophisticated student developed programs which incorporated previously learned BASIC statements and commands to solve assigned problems. Prior to reaching this sophisticated level of problem solving, some students in the program required reinforcement of the BASIC programming language through repetitive sample programs.

The development of higher level thinking skills (enrichment) resulted from the student development of elaborate and original computer programs using unique combinations of BASIC statements and commands. For example, early in the instructional sequence, students were assigned to write a program that would print out a "blastoff" message, followed by a countdown from ten to one, and finally ending in the print-out of a graphically designed rocket. By the end of the five-week session, a participating eighth grade student, had elaborated on this program by having the rocket actually lift off the launch pad! The beginning and ending print-out of the run portion of this program follows: (See Appendix A for actual program).

```
TYPE IN YOUR LAST NAME.?      FORD
THIS IS CAPTAIN FORD.  SERIAL NUMBER 647186
FOR THE U.S.S. ENTERPRISE
WE WILL BE LEAVING FOR THE JUPITER MOONS AT T - 10 SECONDS

* * * * * ALL ABOARD * * * * *

* * * * * STAND BY! * * * * *

10
9
8
7
6
5
4
3
2
1
* * * * * BLASTOFF! * * * * *
* * * * *
```


Enrichment or Acceleration?

```

      ^
     /XX\
    X  X
    X  X
   XXXX
   XXXX
   XXXX
   XXXX
   XXXX
   XXXX
  XXXXXX
 XXXXXXXX
XX  XXXX  XX
XX  'FF'  XX
XX   V   XX
XX   V   XX

```

```

*****
***** LAUNCH PAD *****
*****

```

```

   XXXX
   XXXX
  XXXXXX
  XXXXXXXX
XX  XXXX  XX
XX   FF   XX
XX      XX
XX      XX

```

THIS IS CAPTAIN FORD AGAIN.
 WE HAVE SUCCESSFULLY LAUNCHED AND ARE AT THE ORBITAL
 REFUELING STATION SO THAT WE WILL HAVE
 ENOUGH FUEL FOR THE TRIP TO JUPITER MOONS
 WE WILL BE THERE IN THREE EARTH DAYS

```

***** GOOD BYE *****

```

At the end of the summer session, students were asked to develop an original computer project. The projects ranged from having the computer print out picture graphics, to the developing of programs designed to reinforce basic skills such as math. A fifth grade student in the program, designed a program to help children practice their multiplication facts. The following shows a run of program: (See Appendix B for actual program).

Enrichment or Acceleration?

THIS PROGRAM WILL HELP YOUR MULTIPLICATION FACTS

GIVE ME THE FIRST NUMBER

? 5

NOW GIVE ME THE SECOND NUMBER

? 8

NOW MULTIPLY THE 2 NUMBERS, & GIVE ME THE ANSWER

? 40

CORRECT, THAT'S VERY GOOD!

TYPE 1 TO TRY AGAIN, TYPE 2 TO END

? 1

GIVE ME THE FIRST NUMBER

? 6

GIVE ME THE SECOND NUMBER

? 4

NOW MULTIPLY THE 2 NUMBERS, & GIVE ME THE ANSWER

? 23

INCORRECT, TRY AGAIN

? 24

NOW YOU'RE GETTING IT!

In addition, with the commercialization and popularity of computer games, many of the students in the program developed their own unique and original games. The illustration shows the beginning and ending print out of a basketball game designed by a sixth grade gifted student. The object of the game is to score more points than your opponent. Each player is allowed to choose from two different shot categories. One category is a basic "lay-up" and the other category consists of field goals which differ in difficulty by their increasing distance from the basket. Depending on the program's pre-determined level of difficulty, the computer

Enrichment or Acceleration?

will either print out that the shot is made or missed. Each player has 20 chances to score points before the game is ended and the score is tabulated. (See Appendix C for this program).

```
*****
WHAT DOES THE FIRST PLAYER WANT TO BE CALLED?
? SUNS
THE SECOND PLAYER
? REDSKINS
DO YOU NEED DIRECTIONS?
? YES

***** DIRECTIONS*****

SUNS WILL HAVE THE BALL FIRST.  FOR SUNS 1 = LAY-UP
2=12 FOOTER  3=20 FOOTER.  FOR REDSKINS 4 = LAY-UP
5=12 FOOTER  6=20 FOOTER.  PLAY BALL!
SUNS, WHAT DO YOU WANT TO DO?
? 5
YOU BLEW IT!  YOU TURNED THE BALL OVER.
SUNS, WHAT DO YOU WANT TO DO?
? 3
YOU MISSED!  THE OTHER TEAM HAS THE BALL DOWN COURT.
REDSKINS, WHAT DO YOU WANT TO DO?
? 4
YOU GOT YOURSELF TWO POINTS!

(the game continues until each team has had 20 chances)
*****
SUNS      20
REDSKINS  14
THAT'S THE BALL GAME!  FINAL SCORE SUNS 20  REDSKINS 14
```

An educationally sound program should offer students the opportunity to significantly increase their competency level. Therefore, a pre- and post-test (Minnesota Computer Literacy and Awareness Assessment) developed by the Minnesota Educational Consortium, was administered to twenty-six students who participated in the entire five-week program. Calculation of an analysis of variance showed a significant difference between student pre- and post-test mean scores. (See Table 1.)

Enrichment or Acceleration?

Table 1

Source	DF	SS	MS	F	P
Pre-test	1	2094.2	2094.2	49.83	.01*
Post-test	50	2101.5	42.0		
Total	51	4195.8			

*F= 7.17; 1/50; P= .01

Pre N=26 $\bar{X}$ =28.31 s_x =7.37

Post N=26 $\bar{X}$ =41.00 s_x =5.46

Conclusion:

Educational programs for gifted students should enable them to progress at their own individual rate of learning while simultaneously providing them with the opportunity to utilize higher level thinking skills. As a result of evaluating daily assignments, original programs, and pre- and post-test scores, all students who participated in the T.P.G. computer summer school program were successful programmers at varying degrees of proficiency. According to the 1980 Agenda for Action, the National Council of Teachers of Mathematics (NCTM) recommends computer literacy for all students, at least by junior high and most definitely by high school. Thus, by initiating computer literacy and programming at the elementary level, gifted students have the opportunity for both acceleration and enrichment.

APPENDIX A

```

10 REM SPACE SHIP LAUNCH
20 PRINT: PRINT: PRINT
30 PRINT" TYPE IN YOUR NAME. ";
40 INPUT N$
50 LET H = RND (X)
60 PRINT"THIS IS CAPTAIN ";N$;" SERIAL NUMBER ";H
70 PRINT"FOR THE U.S.S. ENTERPRISE"
80 PRINT"WE WILL BE LEAVING FOR THE JUPITER MOONS AT T - 10 SECONDS"
90 PRINT
100 PRINT
110 PRINT"*****_ALL_ABOARD_*****"
120 PRINT:PRINT
130 PRINT"*****_SIAND_BYE!_*****"
140 PRINT:PRINT
150 FOR X = 10 TO 1 STEP - 1
160 PRINT X
170 NEXT X
180 PRINT"*****_BLAST_OEE_I_*****"
190 PRINT:PRINT:PRINT:PRINT:PRINT
200 LET W = 0
210 PRINT"*****"
220 PRINT TAB(4);"/\"
230 PRINT TAB(3);"/XX\"
240 PRINT TAB(3);"X X"
250 PRINT TAB(3);"X X"
260 PRINT TAB(3);"XXXX"
270 PRINT TAB(3);"XXXX"
280 PRINT TAB(3);"XXXX"
290 PRINT TAB(3);"XXXX"
300 PRINT TAB(3);"XXXX"
310 PRINT TAB(3);"XXXX"
320 PRINT TAB(2);"XXXXXX"
330 PRINT TAB(1);"XXXXXXXX"
340 PRINT TAB(0);"XX XXXX XX"
350 PRINT TAB(0);"XX \FF/ XX"
360 PRINT TAB(0);"XX \ / XX"
370 PRINT TAB(0);"XX \ / XX"
371 PRINT
380 FOR Z = 0 TO 2*Y
390 PRINT
400 NEXT Z
410 PRINT"*****"
420 PRINT"*****_LAUNCH_PAD_*****"
430 PRINT"*****"
440 PRINT:PRINT:PRINT:PRINT
450 LET Y = Y +1
460 ON Y GOTO 240, 260, 280, 300, 320, 340, 360, 371, 470
465 GOTO 550
470 PRINT "THIS IS CAPTAIN ";N$;" AGAIN."
480 PRINT" WE HAVE SUCCESSFULLY LAUNCHED AND ARE AT THE ORBITAL "
490 PRINT"REFUELING STATION SO THAT WE WILL HAVE "
500 PRINT"ENOUGH FUEL FOR THE TRIP TO THE JUPITER MOONS"
530 PRINT" WE WILL BE THERE IN THREE EARTH DAYS"
540 PRINT"*****_GOOD_BYE*****"
550 END

```

Ready

APPENDIX B

```
10 REM MULTIPLICATION PROGRAM
20 PRINT " THIS PROGRAM WILL HELP YOUR MULTIPLICATION"
25 FOR Z =1 TO 2500: NEXT Z
26 PRINT
27 PRINT
28 PRINT
30 PRINT " GIVE ME THE FIRST NUMBER"
40 INPUT M
42 PRINT "NOW GIVE ME THE SECOND NUMBER"
45 INPUT W
50 PRINT " NOW MULTIPLY THE 2 NUMBERS, & GIVE THE ANSWER"
60 INPUT S
70 IF M*W=S THEN 100
80 PRINT "INCORRECT, TRY AGAIN"
90 GO TO 150
100 PRINT "CORRECT, THAT'S VERY GOOD!"
110 PRINT " TYPE 1 TO TRY AGAIN, TYPE 2 TO END"
120 INPUT T1
130 IF T1= 1 THEN 30
140 IF T1=2 THEN 200
150 INPUT S
160 IF M*W=S THEN 180
170 GO TO 80
180 PRINT "NOW YOU'RE GETTING IT!"
190 GO TO 110
200 PRINT "THANK YOU FOR PLAYING THIS GAME"
205 PRINT " GOODBYE"
210 END
```

APPENDIX C

```

1  RANDOMIZE
2  PRINT "*****BASKETBALL*****"
3  LET T1=0
8  LET T2=0
10 PRINT "WHAT DOES THE FIRST PLAYER WANT TO BE CALLED?"
20 INPUT A$
30 PRINT "THE SECOND PLAYER?"
40 INPUT B$
50 PRINT "DO YOU NEED DIRECTIONS?"
60 INPUT C$
70 IF C$= "NO" THEN 160
80 PRINT "*****DIRECTIONS*****"
90 PRINT A$: "WILL HAVE THE BALL FIRST, FOR ";A$;" 1=LAY-UP"
100 PRINT "2=12 FOOTER 3=20 FOOTER, FOR ";B$;" 4=LAY-UP"
110 PRINT "5=12 FOOTER (t+@) FOOTER, PLAY BALL!"
160 PRINT A$;", WHAT DO YOU WANT TO DO?"
165 INPUT Z
168 GO TO 190
170 PRINT B$:", WHAT DO YOU WANT TO DO?"
180 INPUT Z
190 IF Z=1 THEN 240
200 IF Z=2 THEN 260
210 IF Z=3 THEN 280
220 IF Z=4 THEN 240
230 IF Z=5 THEN 260
235 IF Z=6 THEN 280
240 LET D1=INT(RND(X)+.7)
245 GO TO 300
250 LET D2=INT (RND(X)+.8)
255 GO TO 320
260 LET D3 = INT(RND(X)+.6)
265 GO TO 340
270 LET D4=INT(RND(X)+.6)
275 GO TO 360
280 LET D5=INT(RND(X)+.8)
285 TO TO 380
290 LET D6=INT(RND(X)+.7)
295 GO TO 400
300 IF D1=0 THEN 250
310 IF D1=1 THEN 420
320 IF D2=0 THEN 550
330 IF D2=1 THEN 480
340 IF D3=0 THEN 420
350 IF D3=1 THEN 270
360 IF D4=0 THEN 480
370 IF D4=1 THEN 450
380 IF D5=0 THEN 420
390 IF D5=1 THEN 290
400 IF D6=0 THEN 480
410 IF D6=1 THEN 450

```

```
420 PRINT "YOU BLEW IT! YOU TURNED THE BALL OVER."  
430 IF Z > 3 THEN 160  
440 GO TO 170  
450 PRINT "YOU MISSED! THE OTHER TEAM HAS THE BALL DOWN COURT."  
470 GO TO 430  
480 PRINT "YOU JUST GOT YOURSELF TWO POINTS!"  
490 IF Z > 3 THEN 530  
500 LET T1=T1+2  
505 PRINT A$,T1  
506 PRINT B$,T2  
510 IF T1=20 THEN 560  
520 GO TO 170  
530 LET T2=T2+2  
535 PRINT A$,T1  
537 PRINT B$,T2  
540 IF T2=20 THEN 560  
550 GO TO 160  
560 PRINT "THAT'S THE BALL GAME! FINAL SCORE";A$,T1; B$,T2;  
570 END
```

HOW TO DESIGN
EDUCATIONAL MICROCOMPUTER PROGRAMS

Dan Isaacson, Ph.D.
Department of Information Systems and Decision Sciences
California State University
Fresno, CA 93726
(209) 487-2823

If you don't have
time to do it
right the first time,

where will you find
time to fix the
mistakes you made!!

I used to dread debugging computer programs. Back in 1975 I was writing attendance programs in BASIC for Cordova Senior High School in Rancho Cordova, CA. After I wrote a program I would go to the terminal in my classroom at 7am on a Sunday and plan to spend the next twelve hours typing the program into the computer and debugging it. I never seemed to be able to get a program debugged in less than twelve hours.

Then I attended a Saturday conference run by the Northern California Community College Computer Consortium—in particular, a two-hour seminar by the co-author of many books on structured programming concepts, Thomas Cashman. What Cashman had to say made sense, and since I had plans to debug a new program the next day (with the usual dread), I decided to put the ideas I had learned at the conference to the test.

Saturday night I scrapped my existing program and rewrote it with structured, modular concepts. Sunday morning I arrived in my classroom and typed out the program I had designed and written the night before. Five minutes later it was debugged and running! Since then, my debugging efforts have been minimal, meaningful, and non-frustrating.

I'd like to share a technique with you which can't help but reduce your debugging effort to a minimum and which allows you to write complex programs that are easy to read and understand--too often we can't read our own programs after a six month's interval, not to mention programs of other authors we see for the first time.

Designing programs to be used as instructional media tools in the classroom is a planning and writing of educational curriculum combined with techniques used in TV and movies. The six steps which I'll discuss in detail are:

- Statement of the problem.
- Writing objectives.
- Creating a storyboard.
- Writing pseudocode.
- Manual walkthrough.
- Coding and documentation.

Statement of the problem

Too often, the first thing we do when we get an idea is to sit down at our computer and start typing. But fuzzy goals yield fuzzy results and waste much time by forcing many revisions of our plans as we go along. Please refresh your memory right now by re-reading the "proverb" at the beginning of this paper.

As an example for discussion, let's decide to design a computer program which will drill students in spelling.

Writing objectives

Many experiences have been written for computer implementation without regard to whether the exercise can be done more efficiently or cheaply by paper and pencil methods. Just because a computer program can be written to do a job does not mean it should be written. Designing and writing a first rate, well-documented, short program usually takes over 40 person-hours. Program design and effort should be concentrated, therefore, on ideas which make use of the peculiar talents of the computer: ability to constantly adjust for rapid or slow learners, ability to motivate through timing, scoring, immediate feedback, and animation, and ability to summarize student efforts and needs at once without the necessity of distracting a teacher from more important one-to-one teaching duties. The objectives for our spelling program should reflect these computer abilities.

We might like the following features in our program to make use of computer talents:

- a. word flash at varying speed based on word length and accuracy of student.
- b. second try - flash syllables instead of whole word.
- c. third try - display word on the screen until correctly typed.

- d. keep a score based on time and accuracy - like an educational pinball machine.
- e. report number of words correct as well as score.
- f. display list of mis-spelled words at end of game.
- g. teacher selection of words (even Spanish and French).
- h. teacher control of number of words presented in a game.

Creating a storyboard

Once we have our objectives clearly defined we must detail the implementation of those objectives from a pedagogical point of view: exactly what will the user experience? Most program designers are weakest here.

A technique long used by the movie, animation, and TV industries is the storyboard. A storyboard is basically a set of 3x5 cards (or equivalent) tacked to a bulletin board (Figure 1). Each card contains a small sketch of a "scene" indicating whether the scene is a long shot, medium shot, closeup, or extreme closeup, and a written description of the action which occurs in that scene. The cards, and thus the scene sequence, may easily be moved about on the backing during the planning of the project. Cards may also be easily altered, removed or replaced.

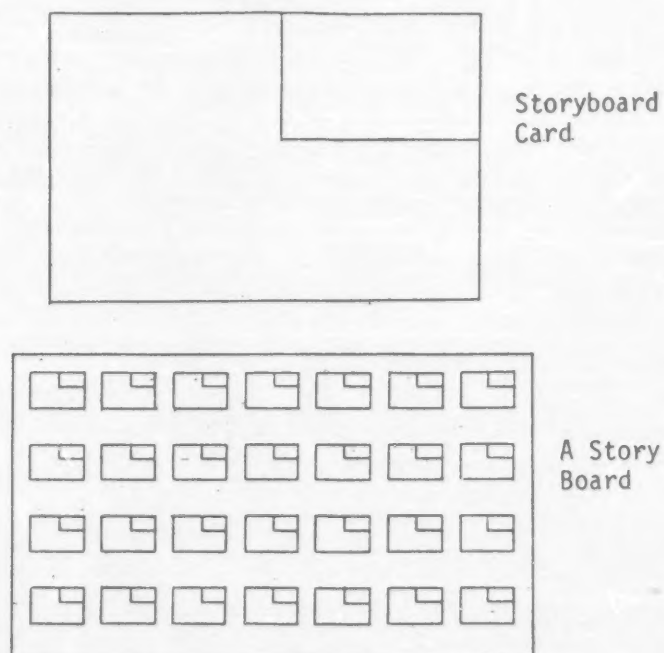


Figure 1

A similar technique should be used in planning a computer lesson. Using graph paper whose width is the same as the number of characters per line and whose height is the same as the number of lines, a carefully spaced layout of the material to appear on the screen is made (Figure 2).

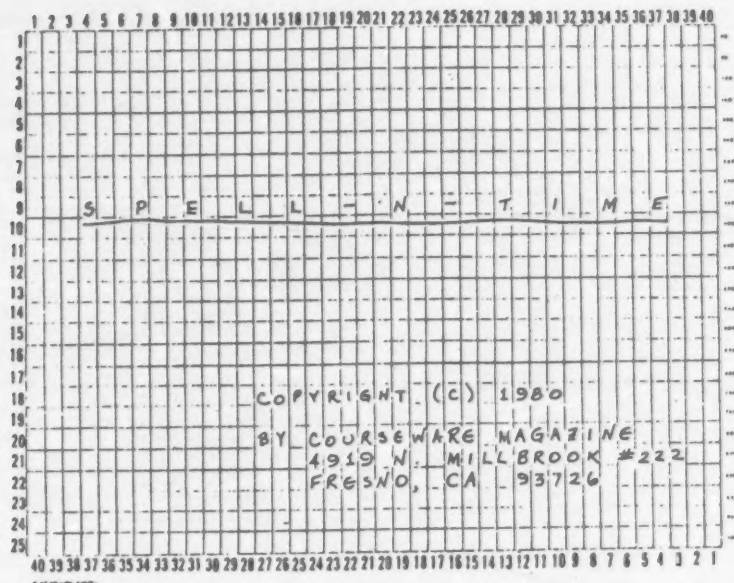


Figure 2

Where various messages can appear on a basic background (Figure 3), overlays are made (Figures 4-7).

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	THIS GAME:										WORDS										TIME:																		
2																																							
3																																							
4	WORDS ATTEMPTED:																																						
5	NO. CORRECT:																				SCORE:																		
6																																							
7																																							
8																																							
9																																							
10																																							
11																																							
12																																							
13																																							
14																																							
15																																							
16																																							
17																																							
18																																							
19																																							
20																																							
21																																							
22																																							
23																																							
24																																							
25																																							
40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Figure 3

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1																																							
2																																							
3																																							
4																																							
5																																							
6																																							
7	WATCH																																						
8	CAREFULLY																																						
9																																							
10	->.....->																																						
11																																							
12																																							
13																																							
14																																							
15																																							
16																																							
17																																							
18																																							
19																																							
20																																							
21																																							
22																																							
23																																							
24																																							
25																																							
40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Figure 4

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1																																							
2																																							
3																																							
4																																							
5																																							
6																																							
7																																							
8																																							
9																																							
10																																							
11																																							
12																																							
13																																							
14																																							
15																																							
16																																							
17																																							
18																																							
19																																							
20																																							
21																																							
22																																							
23																																							
24																																							
25																																							
40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

SORRY, TRY AGAIN.

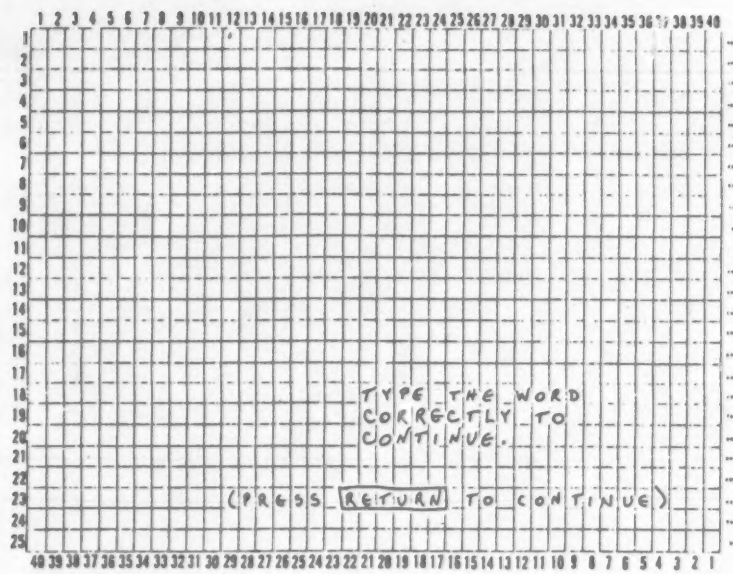
Figure 5

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1																																							
2																																							
3																																							
4																																							
5																																							
6																																							
7																																							
8																																							
9																																							
10																																							
11																																							
12																																							
13																																							
14																																							
15																																							
16																																							
17																																							
18																																							
19																																							
20																																							
21																																							
22																																							
23																																							
24																																							
25																																							
40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

GOOD JOB!!

Figure 6

Figure 7



A conclusion to the "game" is shown in Figures 8 and 9.

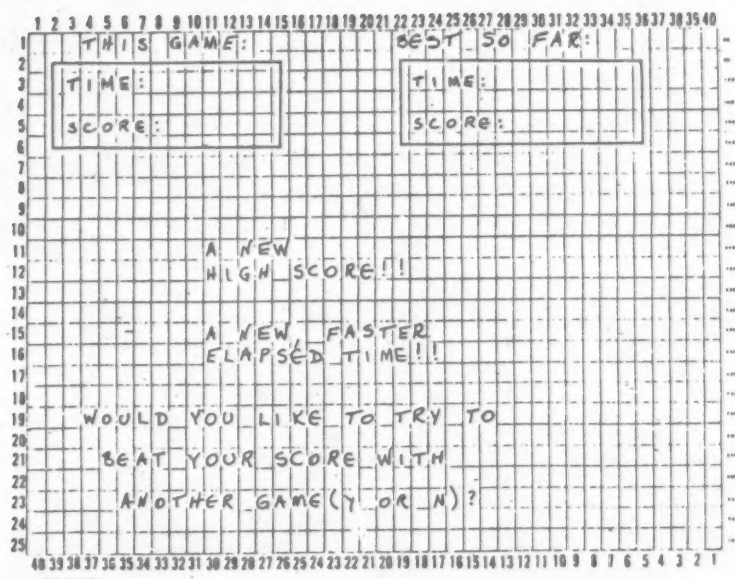


Figure 8

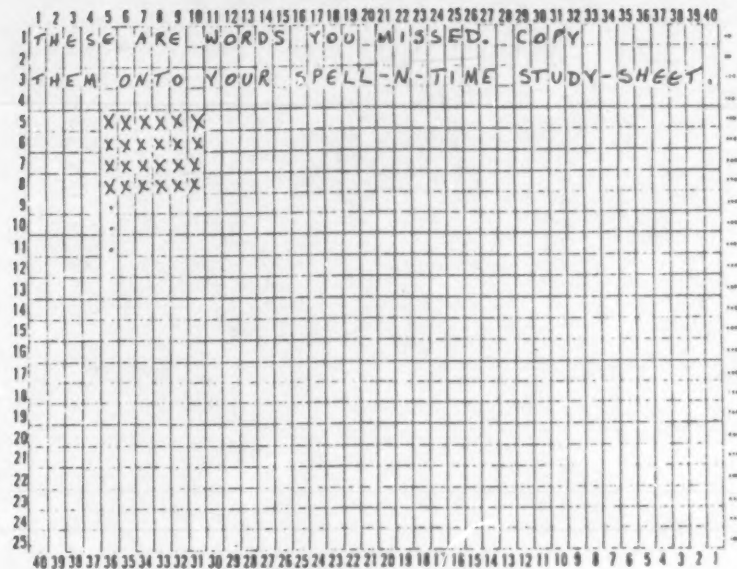


Figure 9

Once the storyboard has been created, the designer should test each objective while viewing the storyboard as if the actual program were being run. It is here that incorrect logic can be found and it is here that a refinement of the student-machine interaction can be made. A program designer should seldom be surprised by a strange screen display with this type of preparation. A designer can also easily add missing experiences at this stage whereas once a program is written, additions are messy.

Writing pseudocode

When the program designer is satisfied that the "frames" give a pleasing appearance and their sequence meets the required objectives, he or she lays out the logic of the program in "pseudocode." Pseudocode differs from BASIC code, or COBOL code, or FORTRAN code in that it is written in plain English logic. A pseudocode program may easily be translated into a target computer language because it is a logically clear statement.

One begins with a gross statement of the problem and then refines it repeatedly until no more refinement is necessary. Thus, the major logic steps for our problem are:

- a. print title page.
- b. play the game.
- c. display the mis-spelled words.
- d. conclusion - comparison of this game to the best so far.

Breaking the major steps into component pieces, we have:

BREAK MAJOR STEPS INTO COMPONENT PIECES:

1) TITLE - STRAIGHT FORWARD.

2) GAME - COMPLEX

A) TITLE PAGE

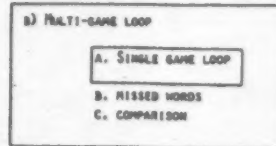


Figure 10

One more refinement of the multi-game loop gives us:

MULTI-GAME LOOP

1. READ VOCABULARY WORDS AND INITIALIZE NECESSARY CONSTANTS.
2. DRAW THE FOUR WINDOWS.
3. PLAY THE GAME.
4. COMPUTE ELAPSED TIME FOR GAME.
5. LIST WORDS MISSED.
6. COMPARISON FRAME.
7. QUERY USER REGARDING ANOTHER GAME DESIRED.

Figure 11

Try to read the final pseudocode logic (or "story") to see how it relates to the storyboards on pages 6-8 in the appendix. At this stage of your understanding it is not necessary to be able to follow every step of the pseudocode logic--we haven't described all the program details for you--but notice the use of indentations, and underlining of if, then, else and while words. The first module, SET START TIME, is a logical whole requiring several, or many, steps in any given programming language, but which can easily be grasped by the reader. Studying a program coded in a specific computer language, on the other hand, often results in questions like, "What are these five lines doing?"

Manual walkthrough

Notice, we still haven't written a single line of code! If we continually debug our logic as we go along, there should be few, if any, bugs in our final coding.

Before coding, once again the program designer should "walk through" the pseudocode testing every possible branch and concept to verify that each module will logically do what it was designed to do. Then, and only then, should coding begin.

Coding and documentation

Coding a program based on pseudocode logic should be done with paper and pencil, not at the keyboard. Each logic module should be coded separately from every other module using modular, structured, top down, one-way-in, one-way-out programming style (which is another whole paper).

Proper documentation inside program code as it is being written must include (see example on p. 11 of appendix).

- many, many remarks.
- "paragraphing" with spaces, horizontal lines and paragraph headings for each module and subroutine
- horizontal spaces to aid readability rather than elimination of all spaces to save memory.

External documentation for educational computer programs should include:

- computer languages available for the program (appendix, page 3).
- subject area covered.
- grade level.
- an abstract of the program.

- basic instructional use.
- how the teacher can personalize the program for his/her classroom (appendix, page 4).
- pseudocode (appendix, page 6).
- description of variables (appendix, page 9).
- an index of subroutines used (appendix, page 10).
- a complete program listing.
- a student guide with student objectives and student worksheet or record sheet (appendix, pages 17, 18).

Summary

In all of the above, no particular computer language was specified. Steps in the design of an educational computer program are language independent and should be so if the final program is to be understood by others than the author of that program. After all, a program may be used as an instructional media tool, or it may be used as a tool to teach programming. As an instructional media tool it must be adaptable to the needs of a teacher who is a non-programmer. As a tool to teach programming, it should be a sample of clarity. We use Shakespeare to teach English, why do less to teach programming?

Suggested reading

Goodman, S. E., and Hedetniemi, S. T. Introduction to the Design and Analysis of Algorithms. New York: McGraw Hill Book Company, 1977. pp. 18-35.

Nevison, John M. The Little Book of BASIC Style: How You Can Write a Program You Can Read. Reading, MA: Addison-Wesley Publishing Company, 1978.

Schneider, G. Michael, Weingart, Stephen W., and Perlman, David M. An Introduction to Programming and Problem Solving with Pascal. New York: John Wiley and Sonce, 1978.

APPENDIX

Statement of Purpose

CourseWare Magazine intends to publish documented, educational programs for the Apple II, PET and TRS-80 microcomputers which may be used by pre-college students at home or in the classroom. Programs will be supplied on cassette tape, in machine readable form, so they may be used right away by students and educators who have no knowledge of programming or skill in typing.

Each issue of CourseWare Magazine will include a C-10 cassette containing at least two programs selected from the ten pre-college curriculum areas of Business, Consumer Economics, English, Fine Arts, Foreign Language, Industrial Arts, Mathematics, Physical Education, Science and Social Studies, or from the area of teacher-assistance programs. Each student program will be accompanied by a teacher guide, a student guide, worksheets (if applicable), suggestions on how to adapt programs for individual lessons, a description of variables used in the program, and a listing of the program. Teacher programs (computer managed instruction, computer supported instruction) will be accompanied by a user guide.

Structured programming techniques will be used in writing the programs and programs will be liberally sprinkled with REM statements and "paragraph" separations.

Fully documented programs take up a lot of memory space; undocumented programs can do fancier things but are very difficult to read. CourseWare Magazine believes that educational programs must be models of the best current program writing style (students learn by example), so we will not attempt to present the fanciest programs possible at the expense of documentation; however, we will present well-designed programs based on educational objectives.

Because school budgets are so tight these days, CourseWare Magazine intends to keep all programs within the limits of an 8K, cassette-loaded machine to reach the largest number of teacher-users; but remember, we are still in the horse-n-buggy stage of educational microcomputer development and it is CourseWare Magazine's intention to provide programs to be studied as well as to be immediately usable educational tools for both the "un-educated" educator and for the "initiated."

Dan Isaacson, PhD
School of Business and
Administrative Sciences
California State University
Fresno, CA 93740

SPELL-N-TIME

Copyright (c) 1980 by
CourseWare Magazine

Computer languages available: Applesoft BASIC 16K
PET BASIC 8K
TRS-80 Level 2 BASIC 16K

Subject areas: All

Grade levels: All

Objectives: After spelling practice with a computer-presented, randomly-selected subset of a set of words chosen by the teacher, the student's computer score will increase and the student's time to complete a spelling exercise will decrease. (Student score is based on spelling accuracy and time.)

Abstract: The computer flashes a whole word; the student must type the correctly spelled word. A second chance is given for incorrect responses, but this time the syllables of the word are flashed in sequence (if more than one syllable). After a second incorrect response, the word is retained on the screen and the student must type it correctly before the game will continue. Score is based on time and accuracy.

Presentation-time of words varies depending upon the ability of the user: a series of incorrect responses will cause slower presentation-time, a series of correct responses will cause faster presentation-time. At the end of each game, a list of mis-spelled words will appear on the screen for the student to write on his/her worksheet.

Basic instruction use: This program is designed as a fun drill in spelling words and in sight recognition of words. It can be used to advantage in any curriculum area where sight recognition and spelling is considered important.

The teacher has control of the vocabulary used and may change or update the vocabulary list daily. The teacher also controls the number of words to be presented to the student during each "game."

Many teachers will create a separate tape for each day's lesson with only the new words for that lesson in the vocabulary list. Spanish or French students (for example) who were absent from a class drill might use the computer review to catch up on what they missed.

Why use the SPELL-N-TIME program? SPELL-N-TIME is a dynamic experience which allows all students to be successful, it makes full use of the computer's unusual talents. Because presentation-time of words in the program is dependent on student skill, weak students are allowed more time to recognize and spell a word, while stronger students are given less time. This permits weaker students to feel success a high percentage of the time because they are correctly spelling words even though their score may be held down because of slower achievement times.

Immediate feedback and high percentage of success is a great motivator--nothing succeeds like success. Neither flash cards nor worksheets have this ability to motivate. You, as teacher, can motivate by personal, one-to-one attention, but you haven't the time, the patience, or the ability to adjust and to keep score that your personal microcomputer has.

How to select the number of words presented in each game: Line 173 (variable: NW) determines the number of words the student will be exposed to in each game. You may change this line by typing:

173 NW = 25

if you wish 25 words to be presented in each game, or:

173 NW = 32

if you wish 32 words to be presented.

Line 173 may be changed by the teacher or user after the program is loaded, or may be changed and the altered program saved on a different tape. (You might title the new program SPELL-N-TIME-32 if NW=32.)

How to insert a teacher-chosen vocabulary: Lines 174 through 198 are reserved for DATA words. You should have at least as many words on those lines as variable NW is set to on line 173. As written, the program will allow for a maximum of 50 words (DIM R\$(50)). To store more words (if the size of your computer's memory will allow), change line 167 to:

167 DIM R\$(60)

or

167 DIM R\$(100)

if you wish to store 60 (or 100) words.

The sample data words are from Spell Correctly, Edition 5, Silver Burdett, 1974. To remove the sample DATA words in preparation for entering your own vocabulary, erase each existing DATA line by typing a line number followed by a RETURN (or ENTER), thus:

174↓ (↓ means RETURN)

175↓

176↓

etc.

Here is a sample vocabulary being added. Type a line number, followed by the word DATA, followed by your DATA words (with a hyphen between each syllable) separated by commas (no comma at the end of a line, though). Put no spaces after commas between words. Thus:

174 DATA dy-nam-ic,ex-per-i-ence,al-lows,all,stu-dents,suc-ce
ss-ful,da-ta

175 DATA be-cause,pre-sen-ta-tio

and so on.

A PET DATA line may be up to 80 characters long (two screen lines). A TRS-80 DATA line may be up to 241 characters long (i.e. about 3 2/3 screen lines), while an Apple II DATA line may be up to 239 characters long.

How to open binder to remove pages for copying: The white, plastic spine may be opened for easy paper removal by simply pulling apart with the fingers. The spine snaps closed by squeezing.

(For PET owners only)

How to fit a 9.2K program into an 8K PET:

You'll notice that the number of bytes used in the program listing is 9231. How can that program possibly fit in an 8K machine?

After writing, entering, running and debugging the completely documented program, we removed every non-essential REM line and REMark. That way, if you need to make changes to the program, your line numbers will be the same as in the listed program. The actual, running program length is about 7K--and therefore fits nicely in an 8K machine.

Procedure

1.0 Introduction and Title Page.

2.0 LOOP

2.1 Initialize all constants and read vocabulary words into array R\$.

2.2 Draw 4 windows and display number of words in game. Press RETURN.

2.3 For words attempted (WA) = 1 to number of words per game (NW)

2.3a set start time (BG) and select a word (W\$),

2.3b remove hyphens from data word and store in variable (FH\$),

2.3c also store syllables in array (FW\$(J)).

2.3d if first try (FL=0)

then store FH\$ also in F\$,
display "watch carefully" and arrow,
flash word,
get response (GT\$),
update time.

if response is correct

then increase score (SC),
add 1 to number correct (NC),
print praise message,
add 1 to consecutive correct counter
(CCC),
set consecutive incorrect counter
(CIC) to zero,
update windows,
press return,
erase answer and message.

else print sorry message,
add 1 to consecutive incorrect
counter (CIC),
set consecutive correct counter
(CCC) to zero,
set attempt flag = 1 (FL),
increment missed-word counter (I),
store missed word (MW\$(I)),
erase answer and message.

```
2.3e  if second try (FL=1)
      then display "watch carefully" and arrow,
          flash syllables,
          get response (G$),
          update time,

      if response is correct

          then  increase score,
              print praise message,
              update windows,
              press RETURN,
              erase answer and message.

          else  print "type the word correctly to
              continue."
              set attempt flag = 2 (FL).
```

```
2.3f  if third try (FL=2)

      then display word (do not erase),
          get response,
          while response is incorrect
              press return,
              erase answer and message,
              get response.
          increase score,
          update windows,
          press RETURN,
          erase flash window,
          erase answer and message.
```

```
2.3g  if word was answered correctly on first try (FL=0)

      then swap word used with last word in array,
          decrement number-of-words counter (N).
```

2.4 Compute elapsed time for game.

2.5 List words missed. Press RETURN.

2.6 Print a frame with time and score comparison to best so far.

2.6a if score is better than best so far

then flash "A new high score!!"
save new high (BS).

2.6b if time is better than best so far

then flash "A new, faster elapsed time!!"
save new elapsed time (BT).

2.7 Query if user wants to try another game.

3.0 END

Description of Variables

BG	start time of game
BS	best score so far
BT	best time so far
CIC	consecutive incorrect counter
CCC	consecutive incorrect counter
ET	elapsed time of game
FHS	flash word without hyphens
FL	attempt flag
FT	flash time
FW\$()	syllables of word
GTS	response
I	missed-word index counter
LP	left pointer
MWS()	missed words
N	number of words in data list
NC	number correct
NW	number of words per game
P5	pause in seconds
R	random number
RS()	random-word array
RP	right pointer
RW	random-word pointer
SC	score
TI	elapsed time
TS	temporary storage
TS	start time
WS	randomly selected word
WA	words attempted

Subroutine Index

7100	update scoring windows
7200	flash words and syllables
7300	timer
7400	erase answer and message windows
7600	compute flash time
7800	watch carefully
8200	pause
8400	'get'
8600	display time
9000	press RETURN

```

10 REM "SPELL-N-TIME"
120 REM =====
122 REM TITLE PAGE
124 REM =====
130 HOME : TEXT
135 VTAB 7: HTAB 9: PRINT "S P E L L - N - T I M E"
145 VTAB 18: HTAB 18: PRINT "COPYRIGHT (C) 1980"
150 VTAB 19: HTAB 18: PRINT "BY COURSEWARE MAGAZINE"
155 VTAB 20: HTAB 18: PRINT "4919 N. MILLBROOK #222"
160 VTAB 21: HTAB 18: PRINT "FRESNO, CA 93726"
165 P5 = 4: GOSUB 8200: REM PAUSE
167 DIM R$(50)
169 REM =====
170 REM WORDS PER GAME &
171 REM DATA WORDS
172 REM =====
173 NW = 10: REM # OF WORDS PERGAME
174 DATA A-BAN-DON, AC-CI-DENT, AC-CUS-TOMED, AD-DI-TION, AD-DRESS, AIL-MENT

175 DATA AL-TI-TUDE, A-MER-I-CAN, AP-PLAUD, A-PRON, AT-MOS-PHERE, AUNT, AU-TO-
-MAT-IC
176 DATA AU-TUMN, AV-E-NUE, AV-ER-AGE, AXLE, BA-CON, BAG-GAGE, BALD, BA-NAN-A,
BAND
177 DATA BAN-NER, BARE, BARGE, BAR-REN, BASE-MENT, BASS, BAT-TLE, BEACH
178 DATA BEAST, BEAU-TY, BEEF, BE-OIN-NING, BE-HAVE, BE-LIEVE, BELT, BI-CY-CLE

179 DATA BLAN-KET
199 DATA "NO MORE WORDS"
200 REM =====
202 REM INITIALIZE
204 REM =====
205 RESTORE
265 R = RND ( - 1 * PEEK (78) + 256 * PEEK (79))
275 SC = 0: I = 0: WA = 0: NC = 0
282 :
284 REM READ WORDS INTO R$(N)
286 K = 1
288 :
290 READ R$(K)
292 IF R$(K) = "NO MORE WORDS" THEN N = K - 1: GOTO 300
294 K = K + 1
296 GOTO 290
298 :
300 REM =====
302 REM DRAW 4 WINDOWS
304 REM =====
310 HOME : A$ = "-----": A1$ = "!" !": A2$
= "-----": A3$ = "!" !"
315 FOR X = 1 TO 6 STEP 5: HTAB 2: VTAB X: PRINT A$ " " A2$:: NEXT : FOR
X = 2 TO 5: HTAB 2: VTAB X: PRINT A1$ " " A3$:: NEXT
320 FOR X = 9 TO 14 STEP 5: FOR X1 = X TO X + 2 STEP 2: HTAB 10: VTAB X1
: PRINT A$:: NEXT : HTAB 10: VTAB X + 1: PRINT A1$:: NEXT
325 HTAB 3: VTAB 2: PRINT "THIS GAME: WORDS": HTAB 26: PRINT "TIME:"
:
330 VTAB 4: HTAB 3: PRINT "WORDS ATTEMPTED:": VTAB 5: HTAB 3: PRINT "NO
. CORRECT:": HTAB 26: PRINT "SCORE:":
410 IF N < NW THEN NW = N: REM FEWER THAN NW DATA WORDS
415 HTAB 14: VTAB 2: PRINT NW:
430 HTAB 36: PRINT "00":
435 GOSUB 7100: REM UPDATE SCORING WNDW
485 GOSUB 9000: REM PRESS RETURN

```

```

500 REM =====
502 REM  SELECT & FLASH WORDS
504 REM =====
507 BG = TI: REM  SET BEGIN TIME          FOR GAME
510 FOR WA = 1 TO NW
515 R = INT (N * RND (1)) + 1
520 W$ = R$(R): REM SELECT A WORD
525 FL = 0: REM  'ATTEMPT' FLAG
530 :
535 IF FL < > 0 THEN 720
540 REM =====
542 REM  REMOVE HYPHENS FROM
543 REM    DATA WORD
544 REM =====
550 FH$ = "": J = 1
555 LP = 1: REM  LEFT POINTER
560 FOR RP = 1 TO LEN (W$)
565 IF MID$ (W$,RP,1) < > "-" THEN 570
567 FW$(J) = MID$ (W$,LP,RP - LP)
568 FH$ = FH$ + FW$(J)
569 LP = RP + 1: J = J + 1
570 NEXT RP
573 FW$(J) = MID$ (W$,LP, LEN (W$) + 1 - LP): REM ATTACHLASTSYLLABLE
574 FH$ = FH$ + FW$(J)
580 REM =====
582 REM    FIRST TRY
584 REM =====
600 F$ = FH$
602 GOSUB 7800: REM 'WATCH CAREFULLY'
605 GOSUB 7200: REM  FLASH WORD
610 GOSUB 8400: REM GET RESPONSE & UPDATE TIME
625 :
640 IF GT$ < > FH$ THEN 680
642 SC = SC + INT (800 / (TI - TS))
643 NC = NC + 1: REM ADD 1 TO # CORRECT
645 VTAB 18: HTAB 23: PRINT "GOOD JOB!!";
650 CCC = CCC + 1: REM  CONSEC. CORRECT CTR.
655 CIC = 0: REM  CONSEC. INCORRECT CTR.
657 GOSUB 7100: REM  UPDATE WINDOWS
660 GOSUB 9000
665 GOSUB 7400: REM ERASE ANS & MESSG
670 GOTO 900
675 :
680 VTAB 18: HTAB 23: PRINT "SORRY, TRY AGAIN.";
685 CIC = CIC + 1
690 CCC = 0
695 FL = 1
697 I = I + 1
700 MW$(I) = FH$: REM  STORE MISSED WORD
705 GOSUB 9000
710 GOSUB 7400

```



```

712 REM =====
715 REM     SECOND TRY
717 REM =====
720 IF FL < > 1 THEN 820
725 GOSUB 7800: REM 'WATCH CAREFULLY'
730 FOR K = 1 TO J: REM     FLASH SYLLABLES
735 F$ = FW$(K): GOSUB 7200
737 P$ = .05: GOSUB 8200
740 NEXT K
745 :
750 VTAB 15: HTAB 20 - LEN (F$) / 2
755 GOSUB 8400
760 IF GT$ < > F$ THEN 875
765 SC = SC + INT (600 / (TI - TS))
770 VTAB 18: HTAB 23: PRINT "THAT'S IT!!"
772 GOSUB 7100: REM     UPDATE WINDOWS
775 GOSUB 9000
780 GOSUB 7400
785 GOTO 900
790 :
800 VTAB 18: HTAB 23: PRINT "TYPE THE WORD": HTAB 23: PRINT "CORRECTLY T
    O": HTAB 23: PRINT "CONTINUE."
810 FL = 2
812 REM =====
814 REM     THIRD TRY
816 REM =====
820 IF FL < 2 THEN 900
825 VTAB 10: HTAB 20 - LEN (F$) / 2: PRINT F$:
830 VTAB 15: HTAB 20 - LEN (F$) / 2
835 GOSUB 8400
840 :
845 IF GT$ < > F$ THEN 875
850 SC = SC + 300
852 GOSUB 7100: REM     UPDATE WINDOWS
855 GOSUB 9000
857 VTAB 10: HTAB 10: PRINT A1$:
860 GOSUB 7400
865 GOTO 900
870 :
875 GOSUB 9000
880 GOSUB 7400
885 GOTO 800: REM     WAIT FOR CORRECT RESPONSE
900 REM =====
902 REM     HOUSEKEEPING
904 REM =====
907 IF FL < > 0 THEN 960: REM     DON'T SWAP IF WORD WAS MISSED
915 : REM     SWAP USED WORD WITH 'LAST' WORD IN RANDOM ARRAY
920 T$ = R$(R)
925 R$(R) = R$(N)
930 R$(N) = T$
935 N = N - 1
940 :
960 NEXT WA
970 ET = INT (TI - BG): REM     ELAPSED TIME FOR GAME

```

```

1000 REM =====
1002 REM  LIST WORDS MISSED
1004 REM =====
1010 IF I = 0 THEN 1100: REM  NO ERRORS
1015 HOME : PRINT "THESE ARE THE WORDS YOU MISSED.  COPY": PRINT "THEM O
      NTO YOUR SPELL-N-TIME STUDY-SHEET."
1025 FOR K = 1 TO I
1030 PRINT MW$(K),: NEXT K
1040 GOSUB 9000
1100 REM =====
1102 REM  TIME-SCORE
1103 REM  COMPARISON
1104 REM =====
1110 HOME : PRINT "  THIS GAME:"; HTAB 22: PRINT "BEST SO FAR:";
1120 FOR X = 2 TO 6 STEP 4: FOR Y = 2 TO 22 STEP 20: VTAB X: HTAB Y: PRINT
      A2$;: NEXT Y: NEXT X
1125 FOR X = 3 TO 5: FOR Y = 2 TO 22 STEP 20: VTAB X: HTAB Y: PRINT A3$;
      : NEXT Y: NEXT X
1137 :
1139 IF BS = 0 THEN 1170
1140 IF SC < = BS THEN 1173
1142 FLASH : VTAB 10: HTAB 10: PRINT "A NEW": HTAB 13: PRINT "HIGH SCORE
      !!!"
1170 BS = SC: REM  SAVE NEW HIGH
1172 :
1173 IF BT = 0 THEN 1200
1174 IF ET > = BT THEN 1210
1176 VTAB 15: HTAB 14: PRINT "A NEW, FASTER": HTAB 18: PRINT "ELAPSED TI
      ME!!"
1200 :BT = ET: REM  SAVE NEW TIME
1210 NORMAL : VTAB 3: HTAB 3: PRINT "TIME:  ";ET;: HTAB 23: PRINT "TIME:
      ";BT;
1211 VTAB 5: HTAB 3: PRINT "SCORE: ";SC;: HTAB 23: PRINT "SCORE: ";BS;
1220 NORMAL : VTAB 21: HTAB 5: PRINT "WOULD YOU LIKE TO TRY TO": HTAB 6:
      PRINT "BEAT YOUR SCORE WITH": HTAB 5: PRINT "ANOTHER GAME (Y OR N)?
      "
1225 GET Q$
1230 IF Q$ = "Y" THEN 200
1235 IF Q$ < > "N" THEN 1225
1290 HOME : PRINT "END OF PROGRAM.  TYPE ";: INVERSE : PRINT "RUN";: NORMAL
      : PRINT " TO START AGAIN."
1500 END
7100 REM =====
7102 REM SUBR - UPDATE SCORING
7103 REM  WINDOWS
7104 REM =====
7110 HTAB 20: VTAB 4: PRINT WA;
7120 HTAB 20: VTAB 5: PRINT NC;
7125 HTAB 34: PRINT SC;
7130 RETURN
7200 REM =====
7202 REM SUBR - FLASH WORDS
7204 REM =====
7215 GOSUB 7600: REM  'COMPUTE FLASH TIME'
7220 VTAB 10: HTAB 20 - LEN (F$) / 2: PRINT F$;: REM  CENTER WORD & DIS
      PLAY IT
7225 TS = TI
7230 IF TI - TS < FT THEN GOSUB 7300: GOTO 7230: REM  FLASH TIME
7235 HTAB 10: PRINT A1$;: REM  ERASE WORD
7240 :
7245 VTAB 15: HTAB 20 - LEN (FH$) / 2
7255 RETURN

```

```

7300 REM =====
7302 REM SUBR - TIMER
7304 REM =====
7310 TI = TI + .1:P5 = .05: GOSUB 8200: RETURN
7400 REM =====
7402 REM SUBR - ERASE ANSWER
7403 REM      & MESSAGE
7404 REM =====
7410 VTAB 15: HTAB 10: PRINT A1$
7415 FOR ZK = 1 TO 3: VTAB 17 + ZK: HTAB 23: PRINT "      "
NEXT
7420 RETURN
7600 REM =====
7602 REM SUBR - COMPUTE FLASH
7603 REM      TIME
7604 REM =====
7610 FT = .05 * LEN (F$)
7615 IF CCC > 2 THEN FT = FT - (CCC - 3) * .1
7620 IF CIC > 2 THEN FT = FT + (CIC - 3) * .1
7625 RETURN
7800 REM =====
7802 REM SUBR - WATCH
7803 REM CAREFULLY
7804 REM =====
7805 HTAB 36: VTAB 2: PRINT "      ";
7810 FLASH : VTAB 7: HTAB 1: PRINT "WATCH": VTAB 8: HTAB 2: PRINT "CARE
FULLY";
7815 NORMAL :P5 = 2: GOSUB 8200: VTAB 7: HTAB 1: PRINT "      " : VTAB 8:
HTAB 2: PRINT "      ";
7820 VTAB 10: FOR ZK = 1 TO 7: HTAB ZK: PRINT " ->";
7825 P5 = .3: GOSUB 8200
7830 NEXT ZK
7832 VTAB 10: HTAB 3: PRINT "      ";
7835 RETURN
8200 REM =====
8202 REM SUBR - PAUSE
8204 REM =====
8210 FOR K6 = 1 TO 700 * P5: NEXT K6
8215 RETURN
8400 REM =====
8402 REM SUBR - 'GET'
8404 REM =====
8413 POKE - 16368,0: REM CLEAR KEYBOARD BUFFER
8415 GT$ = ""
8417 T5 = TI
8420 XQ = PEEK ( - 16384): IF XQ > 127 THEN 8510
8422 IF T5 < = TI THEN FLASH : PRINT " " : NORMAL : PRINT CHR$ (8);
8430 GOSUB 7300: GOSUB 8600: REM UPDATE THE TIMER
8485 GOTO 8560
8490 :
8510 IF XQ = 141 THEN PRINT " " : RETURN
8515 IF XQ < > 136 THEN 8545
8520 PRINT CHR$ (8); " " : CHR$ (8); CHR$ (8);
8525 IF LEN (GT$) < = 1 THEN GT$ = "": REM DELETE FROM STORED WORD
8530 IF LEN (GT$) > 1 THEN GT$ = LEFT$ (GT$, LEN (GT$) - 1)
8534 POKE - 16368,0
8535 GOTO 8560
8540 :
8545 GT$ = GT$ + CHR$ (XQ - 128): REM CONCATENATE NEW LETTER TO WORD
8550 POKE - 16368,0
8555 PRINT CHR$ (XQ - 128);
8560 GOTO 8420

```

```

8600 REM =====
8602 REM  SUBR - TIMER
8604 REM =====
8610 CH = PEEK (36): CV = PEEK (37): REM  SAVE CURSOR LOCATION
8620 VTAB 2: HTAB 36: PRINT INT (TI - TS)
8630 HTAB CH + 1: VTAB CV + 1
8640 RETURN
9000 REM =====
9002 REM  SUBR - PRESS RETURN
9004 REM =====
9010 POKE - 16368,0: REM  CLEAR KEYBOARD BUFFER
9020 VTAB 23: HTAB 7: PRINT "(PRESS ";; FLASH : PRINT "RETURN";: NORMAL
: PRINT " TO CONTINUE)";
9030 X1 = PEEK ( - 16384): IF X1 < > 141 THEN 9030
9035 HTAB 7: PRINT " ";; RETURN

```

SPELL-N-TIME

Copyright (c) 1980 by
CourseWare Magazine

Computer languages available: Applesoft BASIC 16K
PET BASIC 8K
TRS-80 Level 2 BASIC 16K

Subject areas: All

Grade levels: All

Abstract: The computer will flash a word, you are to type the word correctly spelled. If you mis-spell the given word, it will be flashed once more in syllables (if more than one syllable). If your second response is incorrect, the word will be held on the screen until you type it correctly.

Your score is based on speed and accuracy.

Objective: To improve your ability to recognize and spell words. Your score should improve as you play more games.

Use of your worksheet: At the end of each game, the words you are not sure of will be listed on the screen. Copy these words onto your worksheet so you can keep track of your progress, share your progress with your teacher, and practice words you are not comfortable with.

Study sheet

Student _____

Date _____ Class _____

LIST OF WORDS MIS-SPELLED OR INCORRECTLY RECOGNIZED

Game 1

Game 2

Game 3

Game 4

Game 5

Game 6

MICROCOMPUTERS IN THE CLASSROOM

"The education of today's youth is the fulfillment of tomorrow's needs"

Paper presented by Nancy Grimes

Independent School District No. 196 Rosemount, Minnesota

How Can Microcomputers Be Used in the Classroom?

As the object of instruction - Microcomputers can be used to develop computer literacy--an understanding of the application of computers to problems and situations in the academic, business and social worlds. Programs in this area incorporate computer programming, problem solving, applications in various fields, impact upon society and computer system components. By the time District 196 students complete the computer literacy program, they are comfortable, confident and knowledgeable about microcomputers.

As the medium of instruction - Microcomputers can be of great assistance to classroom teachers as tutorial, inquiry and problem-solving devices. Programs already exist in English, physics, chemistry, mathematics, social studies, home economics and many other subjects. Microcomputers can also be used to simulate physical, economic and social systems, and laboratories for science courses.

There are two major categories of use:

- Initial teaching, assisting the teacher as the presenter of information, and
- Learning reinforcement, as a follow-up to teaching that has been presented by the teacher or some other method.

As the manager of instruction - Microcomputers can be valuable assets in test analysis, record-keeping and assigning tasks. They can also be of

help to guidance counselors in the areas of student records, scheduling and research, and to administrators in inventory, budget control and predictive studies.

History of the Use of Microcomputers in District 196

June 1976 - Realizing the potential of computers, District 196, with the assistance of Dr. Norman Bell, a technical expert from Michigan State University, began developing and implementing a computer awareness program for high school students. The purpose of the program was to increase students' awareness of computers and help them understand:

- Computer operations and commands;
- Computer equipment;
- Computer language;
- Computer programming; and
- Computer simulations.

Three basic teaching strategies were planned:

- Teacher introduction and review;
- Hands-on, on line terminal use; and
- Video cassettes.

1976-77 - A four-week course in computer literacy using the time-share system was piloted for ninth grade students. Television tapes and instructional modules that provided hands-on experience were used.

1977-78 - Third, fourth and fifth grade students joined the computer awareness program. Their course included five calculator lessons introducing computing and programming skills, followed by 10 microcomputer lessons to teach BASIC language and programming skills.

1978-79 - A computer literacy program was developed for middle school students.

1979-80 - High school business and science programs were developed.

District 196 Use of Microcomputers in 1980-81

- Focus: Awareness.
- Goals: Computer literacy.
Awareness of impact of computers on society.
Development of computer vocabulary.
- Methods: Hands-on experience with microcomputers.
Mini-lessons with overview, objectives, structured student notes, quick quizzes with immediate feedback, and application problems.
- Media: 10-key, four-function, hand-held calculators.
Microcomputers.
Videotapes, with coordinated transparencies, study guides and quick quizzes.
Handouts.
Teacher presentations.

Computer Awareness Lessons

Overhead visuals and handouts are used in conjunction with the major presentation by the teacher. Each lesson includes an overview, set of objectives, structured notes, quick quiz with immediate feedback and hands-on computer activity.

- Grade 3: Introduction to the Calculator
Errors and How to Correct Them
Mixed Operations, Decimals
Constants and Percents
Using the Calculator's Memory
Introduction to the TRS-80

- Introduction to BASIC
- Introduction to the TRS-80 Keyboard
- Grade 4: Review Lesson 1
 - Arithmetic in BASIC
 - Developing and Storing Programs
 - Branching Statements
 - Looping
- Grade 5: Review Lesson 2
 - Graphics on the TRS-80
 - The READ-DATA Statement
 - Subroutines
 - Numeric Arrays
- Grade 6: Introduction to the Microcomputer
 - Programming in BASIC
 - System and Program Commands
 - Editing
 - Introduction to Branching
 - Introduction to Flowcharting
- Grade 7: BASIC Language Review
 - Extended Arithmetic Operations
 - Numbers
 - Conditional Branching
 - FOR-NEXT Loops
 - Subroutines
- Grade 8: BASIC Language Review
 - Arrays

Nested Loops

READ DATA Statements

String Variables

Advanced Graphics

Grade 9: (emphasis on computer use)

Historical developments

Hardware and software components

Input and output devices

Computer functions

Writing a computer program

Current and future impact of computing systems on society

Computer applications--science, engineering, business,
commerce, education, government

Grade K-12: (other applications)

Elective computer laboratory

Drill and practice in mathematics

Simulations in physics and chemistry

Geography simulations

Energy simulations

Business education projects

Videodisc interface

District Background

Independent School District No. 196 covers 110 square miles just south of the Twin Cities of Minneapolis and St. Paul, Minnesota. All or part of seven cities and two townships are in the District.

As a district in the outer ring suburbs, the District's enrollment has

more than doubled since 1970. Enrollment grew from 11,976 in 1979-80 to more than 12,451 in 1980-81, a gain of 4 percent. According to census projections, growth is expected to continue for several more years.

There are 15 schools in the District--10 elementary (kindergarten through grade five), three middle (grades six through eight) and two high (grades nine through 12).

What is a Microcomputer?

A microcomputer is a small, self-contained computer system that includes the same functional components as larger systems:

- Input--usually by use of a typewriter-like keyboard or a tape player;
- Output--by use of a television screen or printing unit;
- Memory--stores data and programs;
- Microprocessor or central processing unit--handles logical and arithmetical operations, as well as overall control of the computer.

Why Should Microcomputers Be Used in Schools?

There is a critical need for computer literacy in our contemporary world. There has been and continues to be tremendous growth in the use of computers in every facet of American life--business, industry, the home, education, entertainment. Computers are and will continue to be a major force in the lives of Americans and millions of other people around the world.

Computer personnel are in great demand to fill the growing number of positions in this field. Students who have received computer training in school will be able to help meet this need.

Persons who have an awareness of computers will be able to apply their skills in areas that require beginning programming skills, such as real estate, travel, inventory, etc. They will have the ability to determine when the use of a computer is appropriate and have some understanding of the limits and potential of computers to solve everyday problems.

Persons who are computer-literate will have many additional job opportunities open to them, including more complex computer programming positions in business and industry.

Computer professionals will be able to take advantage of almost any opening in the computer field, including research and development with major computer manufacturers. They will also be able to provide valuable assistance to businesses and industries that rely on computers.

For more Information Contact:

Ms. Nancy Grimes

Director of Secondary and Elementary Education

Independent School District No. 196

14445 Diamond Path

Mosemount, Minnesota 55068

(612) 423-4441

Independent School District No. 196 Board of Education:

Kenneth O. Petrick
Beverly J. Albee
Carol Alexander
Donald Westerhausen
Terry Joyce
Jack Ardoyno
Richard Boucher

Chairperson
Vice Chairperson
Treasurer
Clerk
Director
Director
Director

AIM: REMEDIAL MATH FOR SECONDARY STUDENTS

Craig Walker
Resource Teacher
Arrowview Junior High School
2299 N. "G" Street
San Bernardino, Ca. 92405
(714) 886-9118

ABSTRACT:

Arrowview Individualized Math (AIM) is an individualized, diagnostic-prescriptive remedial instructional program for grades 7-12. It consists of 576 prescriptive lessons organized into 32 skill areas, each with pre- and post- tests. The purpose is to help remedial students prepare for proficiency/competency tests. Because AIM is available in a paper format as well, it can be conducted with as few as one or two minicomputers. AIM is available without cost to any school using TRS-80's. A small charge (\$14.50) is made to cover printing costs for schools intending to replicate AIM in the paper format. Schools using AIM report as much as 3 years of growth in computational ability in one semester.

WHAT IS THE PROBLEM?

In a time in which national test scores are falling, teachers are left attempting to prepare students to pass proficiency/competency tests. This is no easy task since as many as 50% or more of students in many secondary schools are below grade level in math.

AIM is an attempt to meet this need and is a powerful tool to help students catch-up and pass required proficiency/competency tests to graduate from high school.

HOW DOES AIM WORK?

AIM is designed to help the secondary student who, for one reason or another, did not master basic math. Thus it is a remedial program teaching skills normally found at grades 3-9. It presently consists of 32 strands or skill areas. Eighteen additional strands are currently being written and will be ready soon.

Each strand is made up of a pre-test, five lessons (each lesson in triplicate in case a student has to repeat a lesson) and a post-test in duplicate, so that if mastery is not obtained on the first post-test, a fresh post-test is later available.

A list of the 32 strands, which is currently ready for replication, is to be found in the appendix. Many high school students have not yet mastered these skills. AIM has thus proven to be applicable to remedial students in grades 7-12.

CHARACTERISTICS OF AIM:

Individualized. Each student is working at a separate place within the strands and at a different rate. Students who are absent can return to school and pick up exactly where they left off. They progress at the speed at which they learn the best.

Diagnostic. The pre-test shows whether or not the student needs any help in the strand. If a student scores 95% or higher the next pre-test is given.

Prescriptive. Up to five lessons can be given in each strand. In the paper format there are three forms of each lesson ("A", "B", and "C") to provide extra work if needed.

In the computer format there is an endless number of forms for each lesson because the problems are randomly generated.

Assessive. Each strand has a post-test which allows the instructor to assess mastery. We suggest 90% as a passing score, but this is flexible. If a student does not reach the passing score, further lessons are assigned, followed by the "B" form of the post-test.

HOW ARE THE COMPUTERS USED?

The entire AIM math instructional program is available on one 50' cassette tape. The computer randomly generates 20 problems within the requirements of each lesson, as well as pre- and post-tests.

The algorithm is not worked by the student on the computer, but rather with paper and pencil. The computer only gives the problem and accepts the answer.

The philosophy behind this operational method is important. AIM was developed for remedial students, none of whom had found success in math. Little would be gained by allowing the student to feel that he/she could only do math on a computer. Rather students are still using paper and pencil and the computer simply serves as the tutor to give the next problem and evaluate the answer.

After 20 problems the computer scores the student. The instructor then calls up the next lesson.

One of the major problems of remedial students is motivation. Computers serve as a powerful motivational device for these youngsters. It is always amazing to see

students with a reputation for misbehavior, industriously working at the computer for the entire period.

AIM adapts well to the number of computers available. Bloomington High School (Bloomington, CA.) uses 30 TRS-80's in one room. An entire class can come in at one time. However, AIM could be used with even one computer in a classroom because the rest of the students could be doing AIM in the paper format. Some schools are using AIM papers only and hope to eventually purchase computers.

The advantage of a computer-based instructional program that can also be done in a paper format is apparent. Wouldn't we all like to start with 30 computers in our school? One or two would be more realistic. The rest of the class can be doing the same instructional program on paper. —

AIM IS AVAILABLE FREE:

AIM is now two years old and was developed using public funds. We are happy to give the computer program to any school at no charge, providing that we are furnished with a 50' or longer cassette tape on which to record it.

The San Bernardino City Unified School District has also prepared replication packets of AIM in the paper format. Each packet consists of nearly 700 sheets (8½" x 11") and includes one copy of every pre-test, lesson, and post-test to used as masters for printing, along with instruction. These packets are available at the cost of printing and preparing (\$14.50).

MINIMUM REQUIREMENTS TO IMPLEMENT AIM:

Computers. As mentioned earlier, AIM can be started without any computers at all and these can be added at some future time. Or even one or two computers can be used during the class period by rotating students for ten or fifteen minute intervals on the computer. They would do the same lessons that they were working on at their seat on paper. More computers could be added as funds become available.

TRS-80, Model I, level 2, 16k machines have been used, since AIM was developed using that brand. Recently other minicomputers are being used such as Apple.

Personnel. The number of adults that it takes to run AIM depends on if the computerized management system is used (see below) or scoring of the papers is done by hand. We have found that students using computers need little attention.

The main requirements for personnel are in running AIM in the paper format. It takes one instructional aide for every 10 students if the papers are being hand-scored. When the computerized management system is used the teacher and one aide can handle a normal classroom.

Material. The largest expense for material is for the printing of the 576 lesson sheets, if they are to be consumed by students not working on a computer. Costs are less if there are enough computers for the whole class (thus no lesson sheets needed) or if students are not

allowed to consume the sheets. Answer strips, developed by Bloomington High School, can be used and the lesson sheets can even be plastic laminated to increase their useful life-span. By September, the additional 18 strands (making a total of 50) will be ready for use. This will, of course, increase the size and cost of the printing.

Equipment. Beside the obvious cost of computers, it is helpful to have some type of enclosure to house them. this is preferable to a table and gives them more protection. We adapt a study carrel (about \$90) for this purpose.

The only other equipment needed is a set of eight literature cabinets to hold the present collection of 576 sheets. When all 50 strands are complete the 900 sheets will require a set of twelve cabinets plus one more to hold the tests (see below) for AIM (other than pre- & post-tests) some of which are still being developed.

CLASSROOM MANAGEMENT OF AIM:

Handscoring. Since multiple choice is used, most sheets can be hand-corrected in about 10 seconds. The results are then entered on the calendar card (see appendix B) for each student. The next sheet is assigned depending upon the students performance on the last sheet.

Computer. A computerized management system has been developed for AIM by NTS Software, that not only corrects each paper but also enters the results in the child's folder. It will then print-out a record of the students work and an evaluation of progress for any period from one

day to one year (see Appendix C).

We use a Scantron model 2012 as the card reader to input. In a short time mark-sense card readers should be perfected that will allow the use of standard IBM data cards. We have one and it is not perfected but does make a neat paperweight.

TESTING TOOLS FOR AIM:

We have not yet computerized a set of testing tools to use to sustain the gain as the student progresses through the strands.

Inventory Test. Gives one problem from each lesson (160 problems). It consists of eight pages (20 problems to a page). Thus one page covers four strands. May be given in stages--page by page--or all together. It is suitable for use as a pre-test or a post-test. It was developed by Don Roberts of Martinez Unified School District.

Checkpoint Test. Gives four problems from each lesson which is 20 problems from each strand, giving a total of 640 problems. It is made up of eight double pages printed on both sides, or 32 pages altogether. It may be used as a pre- or post-test and is designed to be given in stages since darkness would fall before most students could finish wading through a 32 page test.

WHAT KIND OF RESULTS DOES AIM GIVE?

Schools using AIM report large gains in computational skills in a short period of time. Appendix D shows results of students working at Arrowview last fall. It is not

unusual to see gains of 3 years in a semester or less.

PLANS FOR THE FUTURE:

Eighteen additional strands will be available within a few months in paper form. It will take longer to get them in the computer. All software will be upgraded to include graphics to provide added interest.

224

Name _____

Number

PROFILE OF MATH SKILLS

ARROWVIEW INDIVIDUALIZED MATH

This student can correctly:

A large grid for data entry, with columns labeled 'Survey', 'Post-1', and 'Post-2' at the bottom.

1. Identify place value to ten million
2. Add numbers to three digits.
3. Add numbers to five digits.
4. Subtract two digit from three digit numbers.
5. Subtract numbers to four digits.
6. Multiply two digit by one digit numbers.
7. Multiply three digit by two digit numbers.
8. Divide four digit by one digit numbers (no remainder).
9. Divide four digit by two digit numbers (with remainder).
10. Write equivalent fractions.
11. Reduce fractions to lowest terms.
12. Rename improper fractions as mixed numbers.
13. Add fractions with like denominators.
14. Add mixed numbers
15. Add fractions with unlike denominators.
16. Subtract fractions with like denominators.

Survey Post-Test Improvement

A blank grid for plotting data. The grid is 10 units wide and 10 units high. The bottom-left corner is labeled 'Survey' and 'Post-Test' with arrows pointing to the respective axes.

17. Subtract fractions and mixed numbers with like denominators.
18. Subtract fractions and mixed numbers with unlike denominators.
19. Multiply fractions.
20. Multiply fractions by whole numbers.
21. Divide fractions.
22. Divide a whole number by a fraction.
23. Add decimals to thousandths.
24. Subtract decimals to thousandths (without regrouping).
25. Subtract decimals to thousandths (with regrouping).
26. Multiply one digit decimals.
27. Multiply decimals to thousandths.
28. Divide two digit decimals by one digit decimals.
29. Divide three digit decimals by two digit decimals.
30. Rename fractions and decimals as percent and reverse.
31. Rounding whole numbers from ten to one hundred thousand.
32. Rounding decimals from tenths to thousandths.

Survey	Post-Test	Improvement
--------	-----------	-------------



APPENDIX B
REMEDIAL MATHEMATICS

GRADE 7

☐ ABOVE GRADE LEVEL

☐ AT GRADE LEVEL

☒ BELOW GRADE LEVEL

STUDENT: MARK ANTHONY

DATES: 11/3/79 - 11/31/79

INSTRUCTOR: MRS. PROFANT

RED GROUP - PERIOD 4

TEACHER OF RECORD: MR. MATHERS

DATE: DAILY WORK COMPLETED*

11/3 6-4A 85% 6-5A 90% 6 Post-Test A 90%
11/4 TRUANT
11/5 7 PRETEST 25% 7-1B 85%
11/8 DID NOT COMPLETE ANY WORK DISRUPTIVE
11/9 7-2A 95% NO PENCIL TARDY TALKING
11/10 ABSENT
11/11 7-3A 100% NO PENCIL
11/12 7-4A 85% TARDY
11/13 7-5A 50% 7-5B 75% 7-5C 85% 7 Post-Test B 90%
11/14 8 PRETEST 0% TALKING
11/15 8-1A 35% 8-1B 60% NO PENCIL
11/18 ABSENT
11/19 8-1C 90% DISRUPTIVE
11/20 8-2A 80% NO PENCIL
11/21 TRUANT
11/24 8-3A 85% NO PENCIL
11/25 DID NOT COMPLETE ANY WORK TARDY DISRUPTIVE
11/28 8-4A 75% 8-4B 85% NO PENCIL
11/29 8 Post-Test A 85% 9 PRETEST A 45%
11/30 ABSENT
11/31 ABSENT

*EXPLANATION: 6-4A MEANS STRAND 6, LESSON 4, SHEET A

SUMMARY OF PROGRESS FOR THIS REPORT

STRAND 6	PRETEST 35%	Post-Test 90%	Gained 25%
STRAND 7	PRETEST 25%	Post-Test 90%	Gained 65%
STRAND 8	PRETEST 0%	Post-Test 85%	Gained 85%

THIS IS WHAT EACH STRAND TEACHES:

STRAND 6: MULTIPLY TWO DIGIT BY ONE DIGIT NUMBERS.

STRAND 7: MULTIPLY THREE DIGIT BY TWO DIGIT NUMBERS.

STRAND 8: DIVIDE FOUR DIGIT BY ONE DIGIT (NO REMAINDER)

PLEASE CUT-OFF AND RETURN THIS BOTTOM PART TO SHOW YOU HAVE RECEIVED THIS REPORT

STUDENT: MARK ANTHONY GRADE: 7 PERIOD: 4 GROUP: RED

☐ I DO NOT WISH A CONFERENCE AT THIS TIME

☐ I WILL CALL 886-2719 TO SET UP A CONFERENCE APPOINTMENT.

ACHIEVEMENT GRADE D
(FOR THIS REPORT)

EFFORT GRADE D
(FOR THIS REPORT)

CITIZENSHIP GRADE D
(FOR THIS REPORT)

COMMENTS: MARK
NEEDS TO SETTLE DOWN AND WORK
HARDER. CLASS BEHAVIOR OFTEN
REQUIRES CORRECTION BY THE
INSTRUCTOR. DISRUPTIVE 3 PERIODS.

ACHIEVEMENT GRADE IS BASED PRIMARILY
ON AMOUNT OF WORK DONE AND PERCENT
CORRECT. WE THINK THAT MARK
SHOULD DO A MINIMUM OF 2 TO 3
SHEETS EACH DAY.

OUR RECORDS SHOW THAT MARK
WAS ABSENT FROM THE MATH CENTER
6 TIMES DURING THIS REPORTING
PERIOD. WE FOUND THAT MARK
WAS LATE TO CLASS 4 TIMES, AND
CAME WITHOUT A PENCIL 6 TIMES.

WE WOULD BE HAPPY TO HAVE A
CONFERENCE WITH YOU ABOUT THIS
REPORT. CALL 886-2719 BETWEEN
7:30 AM & 2:00 PM TO SET UP A
CONFERENCE APPOINTMENT IF YOU
WISH TO HAVE A CONFERENCE. IF
TRANSPORTATION IS A PROBLEM, WE
HAVE A PERSON WHO CAN BRING YOU
TO SCHOOL AND BACK HOME AFTER THE
CONFERENCE.

APPENDIX C

RESULTS ON ALL STUDENTS TESTED ON DAY OF ENTRY AND EXIT AT ARROWVIEW MATH CENTER*

Sept.- Dec., 1980

<u>Entry Date</u>	<u>Exit Date</u>	<u>Entry Score</u>	<u>Exit Score</u>	<u>Gain</u>
9/15/80	11/7/80	5.1	7.6	2.5
9/15/80	11/7/80	6.0	8.1	2.1
9/15/80	11/12/80	6.0	8.7	2.7
9/15/80	11/18/80	6.1	7.8	1.7
9/15/80	11/25/80	5.3	8.3	3.0
9/26/80	12/4/80	5.5	9.0	3.5
9/15/80	12/12/80	5.8	7.6	1.8
9/15/80	12/12/80	5.0	6.6	1.6
9/15/80	12/12/80	5.1	5.5	.4
9/15/80	12/15/80	7.4	8.7	1.3

*Entry test is the CTBS computation subtest form 3R. Exit test is given after all 32 strands are completed and is form 3Q of the same test. All students are in the 7th or 8th grade.

MICROCOMPUTERS - A NEW TUNE FOR AN OLD SONG

Gena Trimble and Shirley Frye

COMPUTER LITERACY
(for staff)

- A. What is a computer?
- B. How are computers used?
- C. What is the historical development of computers?
- D. How does a computer work?
- E. What are the kinds of computers?
- F. What is this new vocabulary?
- G. What are the uses of computers in education (and in your personal life)?
- * H. Do we need a plan for using computers in the education program?
- I. Would you like some experience using a microcomputer?

*Plan

COMPUTER AWARENESS UNIT

GOAL: To increase an individual's knowledge of how a computer works, what its limitations and capabilities are, and how it affects society.

CONTENT: History of computers
The parts of a computer
How computers work
How to control a computer
What computers can do
Where computers are found
The impact of computers on our lives
Computer activities: states and capitols, logic/games, spelling, math drill

RATIONALE: The suggested content could be included in the 7-8 mathematics courses. In this way all students would be involved at the awareness level.

Hands-on experience is extremely motivating even if used in a limited way. A modest beginning would be to have at least one microcomputer in each 7-8 school.

Our community is rich with resource people who use computers regularly. A good public relations activity would be to have them participate as speakers.

PLANNING: Determine objectives
Identify target population
Prepare for related problems
Conduct staff inservice.

For further information write to:

Scottsdale School District
3811 North 44th Street
Scottsdale, Arizona 85253



SCOTTSDALE PUBLIC SCHOOLS

CHAPARRAL HIGH SCHOOL

6935 East Gold Dust Avenue

Scottsdale, Arizona 85253

Telephone 949-6102

TO: Parents of Computer Programming Students

FROM: Gena Trimble, Instructor.

SUBJECT: Your Student's Enrollment

DATE: August 29, 1980

Your student is currently enrolled in Computer Programming at Chaparral High School. We feel this is a wonderful opportunity for the students to learn the basics of computer programming and how to utilize the computer in problem solving. Due to the complexity of the computer and the cost of using the computer, there are several rules which each student will be expected to follow throughout the semester. These rules are as follows:

1. Students are to use only the terminals provided at Chaparral to access the time sharing system. They are not to use the terminals at the community colleges or any of the other high schools. If a student has the use of a micro-computer available to him, he is encouraged to use it for classroom work to alleviate the traffic on the school's machines.
2. Each student will be assigned his own floppy disc for use on the micro-computers. The care of this disc is the student's responsibility. A student may not at any time use another student's disc.
3. No students are to use the computer terminals after 4:30 p.m. or before 7:00 a.m. All students must adhere to the master schedule for assigned use time.
4. The students are to use the terminals only under supervision.
5. Game playing on the computer is forbidden.
6. No students are to tinker with or move the equipment in the math lab without the permission of Mrs. Trimble.
7. Any violation of these rules would result in suspension from the course.

We would appreciate your signature below to acknowledge receipt of this notice and to grant your student permission to take this course. We are looking forward to having your student in class this semester.

(Student's Name)

(Parent's Signature)



SCOTTSDALE PUBLIC SCHOOLS

CHAPARRAL HIGH SCHOOL
6935 E. GOLD DUST AVENUE

SCOTTSDALE,
ARIZONA 85253

TELEPHONE 949-6102

TO: Computer Programming Students

FROM: Mrs. Trimble

RE: Computer Programming Class Rules

1. Class will meet formally two $\frac{1}{2}$ hour sessions per week unless otherwise specified.
2. Each student will have a time of $1\frac{1}{2}$ hrs. per week assigned to do terminal work.
3. Students will notify the teacher one day prior to an assigned time of any needed temporary schedule change. All missed times must be made up.
4. Violation of any of the following rules will result in immediate expulsion from class.
 - a. No food or drink in lab.
 - b. No deliberate vandalism of lab equipment including walls, chairs etc.
 - c. No tampering with anyone's discs or files, or other personal materials.
 - d. No presense in the computer room beyond assigned hours.
7:00A.M.--3:30P.M.
 - e. No hogging the machines beyond individual assigned time.
 - f. No intimidation of other students.
5. Grades will be based on projects, classroom cooperation, and exams over lecture material.

I _____ agree to abide by the above rules.

Student Signature _____ Date _____

SOME ELEMENTARY PROGRAMS FOR BEGINNERS:

LOGIC:

1. Triangle Inequality:

Write a program which will determine whether three lengths can possibly form a triangle. If so determine if the triangle is isosceles, equilateral, or scalene.

2. Equations of Lines:

Write a program which will determine the equations of two lines when two ordered pairs for each line are given. It should tell if the lines are vertical, oblique, or horizontal. Also whether they are parallel or intersecting and, if intersecting, whether they are perpendicular or not. The program should print out the equations of the lines together with all the information regarding their relationship.

3. Roots To Quadratic Equations:

Write a program to find the roots to a quadratic equation. It should accommodate both real and complex roots. Use a formatted output to print out the equation together with the roots. (Hint: Test the discriminant and then use the quadratic formula.)

LOOPING:

1. Factorials:

- Write a program which when input any positive integer N, will compute $N!$.
- Write a program that will generate Pascal's Productive Pattern using combinatorial analysis.

$$\binom{n}{r} = \frac{n!}{(n-r)!r!}$$

2. Computing PI:

Leibnitz showed that the value of one-fourth pi was equal to an infinitely long series:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots$$

Each fraction in this series is called a term.

Use this series to compute the value of π correct to one decimal place. How many terms of the series does the program use?

Use the series to compute the value of π correct to two decimal places, and once again to three decimal places. Your output should be a table with the headings:

ACCURACY(decimal places)

PI

#OF TERMS USED

3. Primes:

- a. Write a program that will find all the prime numbers from 1 to some positive integer N.
- b. Write a program that will determine if an integer is prime or composite and if composite will prime factor the integer.

SUBSCRIPTED VARIABLES:

1. Counting Votes:

It is possible to write a program which will count the votes in an election, or tabulate the responses to a questionnaire, or tally any other list of numbers. The input data for such a program might be a sequence of one-digit numbers, where "3" would represent one vote for candidate Number 3.

Write a program which, when given any list of one-digit numbers, will tell how many 1's there are in the list, how many 2's, and so forth. When the program reaches the end of the list, it should also print the total number of numbers tallied. Use a random number generator to produce a list of integers.

2. Cramer's Rule:

Write a program which will solve any three simultaneous linear equations.

EXAMPLE ASSIGNMENT:

SEARCHING:

Write a program which will generate and store a list of 999 random integers between 1 and 999, will ask the user what number to search for, and will then print the number of times that that number appears in the list.

Discussion: In order to search easily, the list should be sorted. An exchange sorter will not handle a list this long (it would take about one million passes through the list.) However, sorting by distribution can be used. Establish a counter for each possible random integer X. Then every time an integer is produced by the random-number generator, increment its counter by 1.

THE ALPHASYNTAURI; A KEYBOARD BASED DIGITAL PLAYING
AND RECORDING SYSTEM WITH A MICROCOMPUTER INTERFACE

By Charlie Kellner and Ellen Lapham, Syntauri Ltd., Palo Alto
and Laurie Spiegel, composer, New York City

abstract:

Interfaced to an inexpensive, readily available microcomputer, the alphaSyntauri™ provides musicians and studio technicians with a digital synthesizer. It can be used for sequencing and real time effects, timing, repeats, and timbre. Waveforms and timbres may be created and stored on disk using built-in additive synthesis and analysis programs.

Introduction

The alphaSyntauri is a modular, general purpose microcomputer based instrument intended for music composition and real time performance. The philosophy which guided the system's designer, Charlie Kellner, was based on one fundamental concept:

A flexible, useful instrument can successfully use software for real time instrument operation and for generating and controlling timbres and effects.

Background

Historically, sound synthesizing devices have been hardware based. The programming of early synthesizers was accomplished using either hard-wired equipment or patch bays, such as in the original Moog synthesizer.

Dependence upon circuitry and hardware has inherent limitations. First, the 'programming' is not easily changed. Patch bays can become complex, and on-the-road modifications require hardware expertise, time, and patience. Second, the cost of the instrument increases as a function of size and complexity - so expansions and modifications create cost penalties, and thus have tended to be limited to large, fixed installations.

Today, digital technologies are opening horizons for music exploration and control. Uses of digital technologies for sound synthesis range from dedicated microprocessors to full general purpose computer based systems. Microprogrammed synthesizers have been effectively 'hard wired' at the software level: the user typically does not have access to the microprocessor to make programming and control changes. (The Prophet 5 is an example of the microprogrammed synthesizer.)

Taking advantage of computer system technologies, the Fairlight and Synclavier, for instance, use more sophisticated techniques such as disk storage media, and allow a greater degree of user control.

The alphaSyntauri uses a general purpose computer system, the Apple II- which is designed simply for information handling for any purpose. Here, the system software designer puts control in the user's hands through software commands, letting the user modify - or even create from scratch- his own musical instrument.

Controlling and accurately replicating sounds and note sequences is accomplished through programming (software). A user can decide, for

Syntauri Ltd.
AES/Nov. 1, 1980

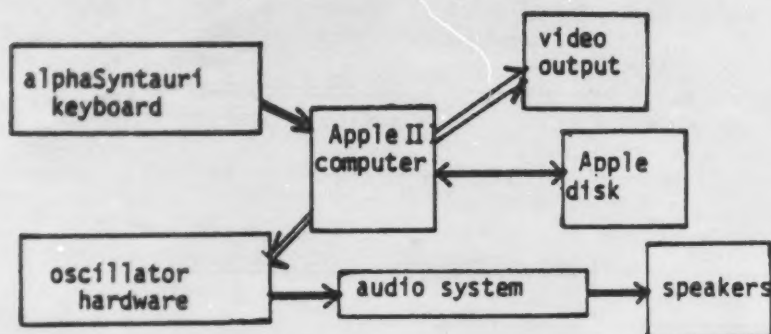
instance, to additively synthesize ten timbres by using a built-in software module, the additive synthesis program. Saving these timbres to diskette is accomplished using standard Apple II disk utilities called by the program. Once saved, the new instruments are stored in unambiguous digital form.

It is the general purpose nature of the computer which here provides new options to the music industry when working with 'synthesized' sounds. From the instrument definition process to the editing of recorded pieces, the Apple II supports the user with a multitude of software tools, languages, printers, and utilities.

Users may send programs, instruments, and even recorded pieces over phone lines using conventional Apple II peripheral equipment. The computer is completely impartial and accurate - thus an improvisation recorded in Los Angeles can be sent to an alphaSyntauri site in New York along with the instrument definitions and effects for review and editing or mixing.

The alphaSyntauri system structure

Primary system components are the alphaSyntauri keyboard, software, and interface hardware, plus the oscillators (available from various sources); the Apple II computer system; and the audio system. These system components are illustrated in the accompanying diagram.



The system software

The major alphaSyntauri performance program, called ALPHA III, is a software-based real time emulation of the traditional synthesizer. The program performs all the functions of a synthesizer with hardware such as switches, knobs, patch cords and hard circuitry such as envelope generators, mixers, and filters.

In operation, the program calculates what a hardware (analog) synthesizer would be doing every 20 milliseconds, and controls the output oscillators in real time to achieve this. Because many major features of traditional synthesizers have been incorporated into the ALPHA III program, the processing cycle is complex.

To achieve the required musical results from the alphaSyntauri system, the languages with which the system has been written have been chosen for their specific benefits. The programs embedded in the process loop are written in 6502 native assembly code for speed. The main control program which sets up the loop and allows user-selected parameters to control the cycles, is written in BASIC, a well known and easily modified programming language.

Syntauri Ltd.
AES/Nov. 1, 1980

The alphaSyntauri process

Within the alphaSyntauri system, there are multiple processes (or tasks) being done essentially simultaneously. The computer is sufficiently fast that all these tasks, and their attendant decisions, are executed without impairing the sound quality.

System tasks include reading the keyboard input device, determining which key was pressed, reading tables (envelopes, waveforms, etc.) and updating the output oscillators. See the attached process flow diagram.

The oscillator update process is the most fundamental process in the alphaSyntauri system. The update tells the output hardware to produce a sound at a certain volume and frequency, until the next update cycle, when the oscillator control(s) may be changed to reflect new keys pressed, new envelope stages, or the termination of the note.

A keyboard event...the pressing of a key during live play...initiates many processes itself. This event, interestingly enough, is treated as an exception event, as its detection and determination of what to do with the information takes less than 1% of the total process loop time.

Examples of software-controllable parameters and features

The following examples were selected to demonstrate the range of features and control options achievable using the general purpose micro-computer, the Apple II. Extensive use is made of the Apple's utilities, memory handling, and disk operations to extend the alphaSyntauri instrument beyond what would have been possible using dedicated electronic components.

The tradeoffs involved in not selecting possibly higher speed dedicated microprocessors have not materially impacted the audible result: in fact, it is the generality and expandability of the system design which is achieved only through the use of a low-cost commercial computer system.

Keyboard tuning

The alphaSyntauri has been designed, as standard for the first release, to have 12 keys per octave, with a minimum resolvable step of one quarter tone. Thus, there are 24 slots in a pitch translation table (loaded into RAM) which are available for frequency assignment for one octave.

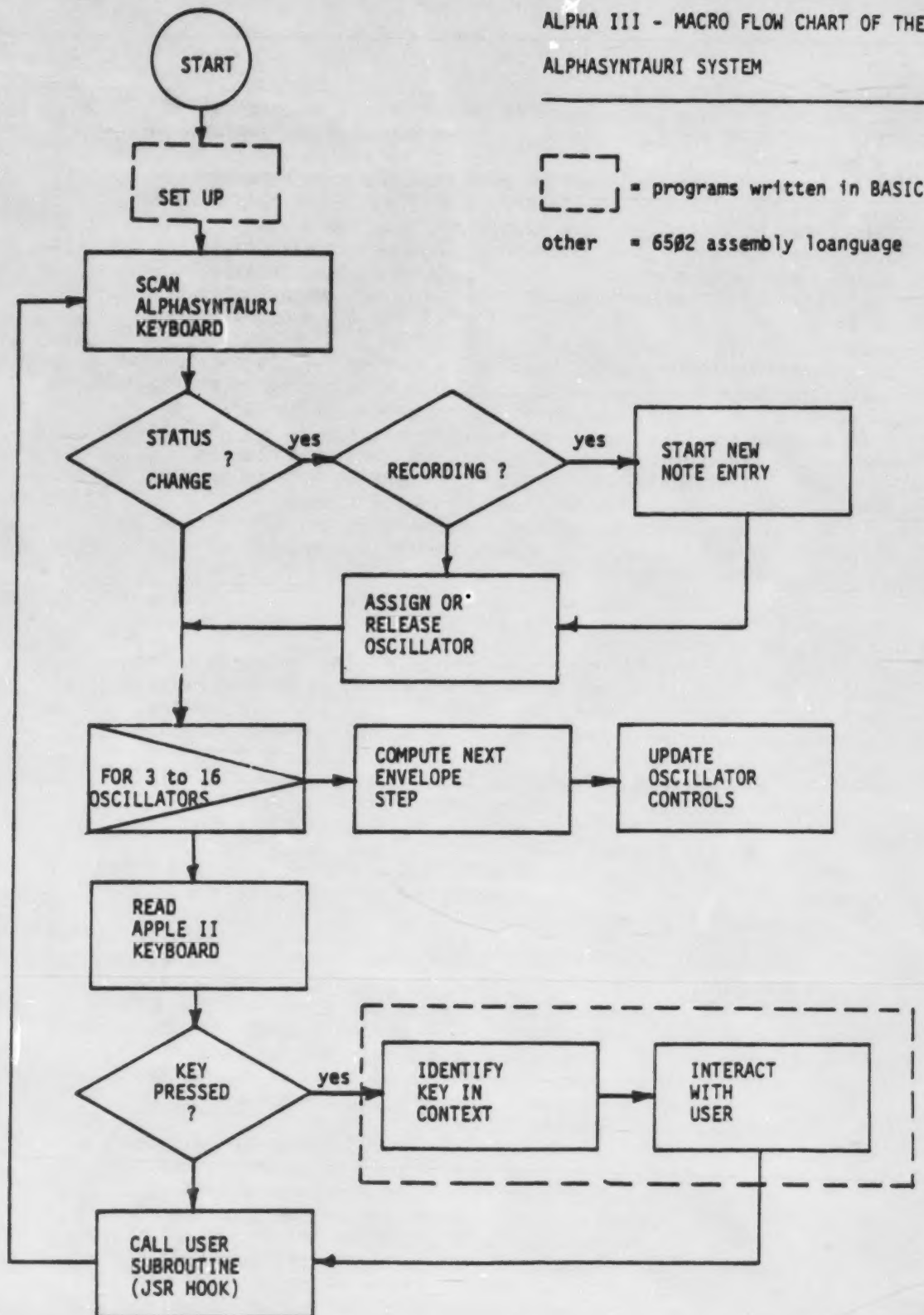
To minimize table size, the system assumes that frequency doubles with each octave, so that the frequency of one octave can be derived by multiplying (or dividing) by two. Computers are especially proficient at this task.

During real time performance and playback, the keyboard can be dynamically tuned using the FC (frequency control) software from the Apple II keyboard. Users may input any number from 0 to (effectively) 180 or so: middle C corresponds to 30 in mid-keyboard, 6 transposes the keyboard down one octave, still in the key of C).

For finer tuning adjustments, the user exits real time control to recalculate the entire table from which the oscillator frequency assignments are made. The tuning process is: 1) RUN SCALE III program*, 2) LIST lines 0 to 99, and 3) change the appropriate parameters in the internally documented program to fine tune to within nineplace decimal accuracy.

* currently, three types of scales, well tempered, just, and international, are built in to the alphaSyntauri.

ALPHA III - MACRO FLOW CHART OF THE
ALPHASYNTAURI SYSTEM



Software features, continued.

Variable Speed Playback

During the playback loop, there is an additional process called READ APPLE GAME PADDLES. Because of the way they are implemented in hardware, it takes from zero to approximately 20 milliseconds, in a linear fashion, to read the paddles. The read time simply increases the process loop linearly according to the paddle setting. Thus, during a 'read' of the paddle setting, the process loop time increases: the performance/playback process can be slowed down by as much as 50%, resulting in a slower playback.

The variable speed does not affect the instrument definition or in any way affect the sound quality of the piece being played back.

Faster playback is also available. During the playback process, a duration byte which was attached to each note during recording, is counted out in the same way it was recorded. (This duration byte is essentially a length-of-real-time-passage counter.) Pressing the Apple II game paddle 'target' button initiates a fast forward by causing the duration byte to be counted out by twos. This doubles the playback speed.

Normal variable speed playback range is 50 to 100% of the original input (playing of the keyboard) speed, and fast-forward doubles this range from 100 to 200%.

Analysis of waveforms

Waveforms are user created through additive sine synthesis. The program ANALYZER III allows any waveform stored on disk to be separated into its harmonic components. A simple digital filter algorithm is executed for each of the first twenty harmonics (more if desired). The resulting display reports their relative amplitude on an arbitrary scale (roughly from 1 to 200.)

The harmonic analysis can be used, of course, with the built-in WAVE III synthesis program to re-create the analyzed waveform. A more interesting use might be to process the first waveform through a filter program to obtain the equivalent filtered waveform. Two waves could be added harmonically to obtain a third. (All it takes is a little imagination.)

Velocity sensing

Each key on the alphaSyntauri keyboard has two electrical contacts, one which makes about one third of the key down travel, the other which makes at bottom. When a key is pressed, the following occurs:

1. Upper contact closes. Here, a count register in the Apple II's memory which is assigned and maintained for each key is set to zero. No other action takes place.
2. The lower contact closes. The count value which has accumulated in that key's timing register is used as an index into the velocity assignment table. The attack rate and attack target volume of that oscillator are here determined. (note, the psychoacoustics of perceived effects as a function of key velocity have been handled by varying both the attack rate and target volume, where loudness is perceived

- to increase not only with absolute loudness increases, but also with the quickness of getting there.)
3. A new oscillator is then assigned for the key which has been pressed.

To achieve velocity sensing results, attack rate and volume are both used. Typically, the attack rate and volume for a given key are inversely proportional to the time between contact closures. That is, velocity is the inverse of transition time. The actual volume effects, changes to attack rate and volume, are handled by a look up table which is loaded automatically by the ALPHA III software into a specific memory location.

Now, to obtain a stiff or loose keyboard, a parameter value is set in the program which effectively makes the table values more or less linear. To illustrate:

The linear, loosest keyboard has a parameter value of 7.99

The logarithmic, stiffest keyboard, has a parameter of 0. In the stiffest keyboard, the key has to be struck very fast and sharply to get loud volumes.

The table values used affect both the real and perceived energy differences between a soft or a hard (fast) key depression. For a soft key stroke, there is less energy in the sound. That is, a lower target volume is arrived at much later. It is the first few milliseconds of hearing which determine the perceived energy in the sound, so the note that reaches peak volume first has the most energy.

Flexibility in the velocity sensing results can be further achieved through completely reworking the look up table from which the velocity sensing results are ultimately derived. In addition, being a general design, the parameter(s) of the envelope which are affected by the velocity sensing process may be altered from the attack rate/target volume to, for instance, the attack target volume/decay rate.

User's own subroutine

The main process control loop (see illustration) contains what is called an unconditional JSR (jump to a user-written subroutine). This allows any user familiar with 6502 assembler code to devise his/her own special effects and controls - which are always assessed during the process cycle.

In keeping with the goal of producing realistic and pleasant sound effects, the time taken to perform the user's subroutine should be kept under one millisecond (1000 microseconds).

An example of a JSR might be to process the note information through a table for frequency modulating the sound.

More information may be obtained by contacting Syntauri Ltd, 3506 Waverley Street, Palo Alto Ca 94306. (415) 494-1017.

Syntauri Ltd.
AES/Nov 1, 1980

Frequency Modulation : a modification to the alphaSyntauri music system.

by Laurie Spiegel

In order to provide a means of generating more complex timbres and repeating patterns of pitches from a single keypress, I implemented a software modification which permits the use of a form of digital frequency modulation to achieve these effects. Using dynamically changing variables already present in the system, this approach to more varied musical output is highly economical, adding little time or code to the system.

Various (and varying) values are added or not added to the pitch value currently to be outputted to hardware to provide this modulation. These values are drawn from addresses, storage locations where dynamically changing system variables reside in memory. (For example, the location of one byte of the instantaneous amplitude of the voice whose frequency is to be modulated.)

The address of the storage location from which the frequency offset is to be taken may be changed from the BASIC control program level through a 'poke' command. During any user-triggered (ASCII entry from the Apple keyboard) pass through the BASIC level program, the second of the two values required to produce the modulation is read from one of the Apple's game paddles. This value, an "FM mask", is stored and used to generate the modulating signal by being "AND'ed" with the dynamically changing variable discussed above. The result of the AND is then added to the current keyboard-derived pitch value as a varying offset, producing the frequency modulation.

The range of effects possible with the FM modification to the alphaSyntauri includes the following:

Use the high byte of the current voice's instantaneous amplitude:

1. No effect, normal alphaSyntauri operation, if value read from knob input is zero.
2. Continuous glissando changing of frequency in the same pattern as the envelope (similar to a common hardware analog synthesizer patch) if the knob value is 255.
3. Frequency modulation by any of 253 possible knob-selectable complex waveforms which result from sequences of logical AND's between the knob-defined mask and the changing envelope as updated by the envelope routine for the instrument currently selected.
(This affords different timbres and effects for different envelope 'presets' too, and constitutes one reason why I originally implemented the bank of ten instrument presets loaded into RAM, as opposed to the single one which had previously been resident.)
4. Any of the above effects, but slowed down in tempo into the melodic rather than timbral range. This is achieved by holding down any ASCII key and the Apple's 'repeat' key, thereby forcing a full loop through the BASIC program to the introduced between changes of pitch which are produced by the frequency modulation code.

Due to the polyphonic nature of the instrument, dense textures of such timbres and patterns may be easily created, permitting the alphaSyntauri system to be used for musical effects more characteristic of synthesizers than of more conventional keyboard instruments. This textural and timbral potential provides a rich area of exploration which can be utilized and enjoyed with or without the use of conventional keyboard technique.

The wide range of completely replicable (though difficult to predict) effects are achieved quite economically as this FM modification requires only two additional parameters of control, and adds 26 bytes of 6502 code and two BASIC statements to the alphaSyntauri system.

Syntauri Ltd. AES/11-1-80

THE USE OF MICROCOMPUTERS IN MUSIC INSTRUCTION

David Cohen
Department of Music
Arizona State University
Tempe, Arizona 85281

The following is not an "abstract," but rather a description of an informal "show and tell" session.

Preliminaries: Microcomputers can deal with factual material in music (for example, composers' dates and works, definition of terms, chord spellings) exactly as well as they can deal with similar material in any other field.

Product review: This dealt only with music systems available for the Apple II. There are three main plug-in boards which give the Apple the capability of producing programmable multi-voiced compositions. The A.L.F. system has superior graphics and a sophisticated system of dealing with declared meters. (For example, if the user tries to program too many beats into the meter he has selected, the program will tie the proper note values over the bar line.) The limitation of this system is that the tone color is, for most uses, only square waves. Each board makes available three voices, up to 5 boards which the software can handle. The system with the most potential (not yet fully realized in the company's software) is the Mountain Hardware Music System. The company promises fully programmable 16 voice capability, with user defined waveforms and envelopes employing digital synthesis which can achieve fidelity approaching 15,000 cycles per second. This is quite an achievement, and when realized will make the Apple II a digital synthesizer which does a great deal of what commercially available digital synthesizers costing 5 to 8 times as much can do today.

As an instructional tool in the development of aural skills (as

well as drill in "information"), the best available system is the series of programs developed by Micro Music Corp. Neither of the previously mentioned companies have done any serious work in CAI. The Micro Music "Music Composer" plug-in board does not require an auxiliary audio system (the board has its own amplifier) and employs digital synthesis, with user defined timbres. Its limitations are that fidelity is limited to about 3500 cycles per second and moving inner voices will interrupt sustained upper voices. At present, however, this company is years ahead of anyone else in developing tested, pedagogically sound and effective programs in music instruction. The software support they provide for their hardware board simply can't be matched (and won't be, in my opinion, for several years to come.)

The rest of the session: Volunteers from the audience were encouraged to try out two of the Micro Music programs--one aimed at 3rd to 8th graders (or adults who might feel themselves to be at that musical level) and another which attempted to teach pitch (or scale number) recognition at a level constantly adjusted by the computer to match the capabilities of the user. From comments received after the session, this demonstration appears to have convinced many people of the effectiveness of computer assisted instruction in the development of aural skills at about every level.

Note: A brochure with detailed descriptions of the Micro Music instructional programs may be obtained from: Micro Music Inc., P.O. Box 386, Normal, IL 61761.

INSTRUCTIONAL TECHNIQUES in TEACHING
BASIC PROGRAMMING to ELEMENTARY CHILDREN

by

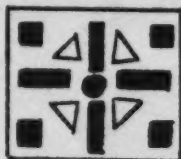
Marilyn Sue Ford
Clark County School District
2234 Pommel Avenue
Las Vegas, Nevada 89119

GOAL: Development of Logical Thinking Skills

OBJECTIVE: Given geometric designs (in this example, created from four shapes and four colors), the student will develop logical thinking skills, a prerequisite to computer programming. Given the opportunity to create designs from oral directions, the student will come to realize and analyze the need for logical and precise directions.

MATERIALS: Several pre-made designs on cardboard.

Example:



One packet of geometric shapes per student. Approximately six of each shape and color.

PROCEDURES:

1. Teacher directed.

A. Using a design card unseen by the students, the teacher orally presents precise, logical directions. As each set of directions is presented, the students create the design the teacher is describing. After all directions are given, the students compare their designs to the design card.

B. Using a different design card unseen by the students, the teacher orally presents unclear, illogical directions. Once again, the students attempt to create the design the teacher is describing. After all directions are given, the students compare their design to the design card.

C. Discuss the two different approaches used by the teacher for creating a design from directions. Elicit student responses as to which method was easier and why. Develop the concept that precise, logical directions are the most efficient. (Note: Working from the center outward is usually the more logical approach.)

2. Student directed.

Over a period of time, each student is given an opportunity to play teacher by orally presenting directions to their peers. As the students become more and more proficient, change the procedures as follows:

A. Student presents directions to the class while looking at the design card.

B. Student is allowed to look at the design card for one minute and then presents directions to the class from memory.

C. Two students leave the room with a design card in an envelope. While looking at the design card, student one describes the design to student two. After rejoining the class, student two presents directions to the class.

D. Student creates own design for presentation to the class.

211

DEVELOP FLOW CHARTING SKILLS

Flow Chart Defined - A flow chart is a road map or drawing of all the steps used to solve a problem. It shows the order of the steps and how each step is connected to the other steps.

A. Start with a list of unordered things to do.

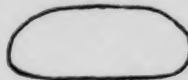
1. Get up
2. Put on shoes
3. Go to school
4. Put on socks
5. Eat breakfast

B. Put the list in correct order.

1. Get up
2. Put on socks
3. Put on shoes
4. Eat breakfast
5. Go to school

C. Introduce these flow chart symbols.

Start or stop



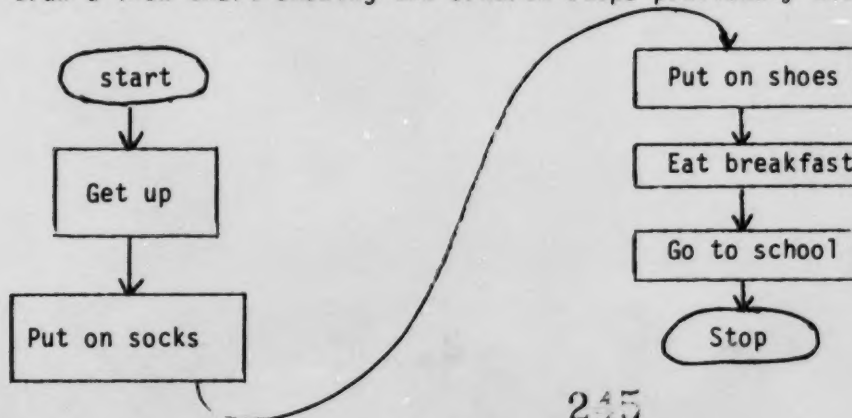
Instruction to do something



direction arrow

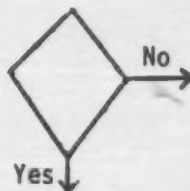


D. Draw a flow chart showing the ordered steps previously discussed.



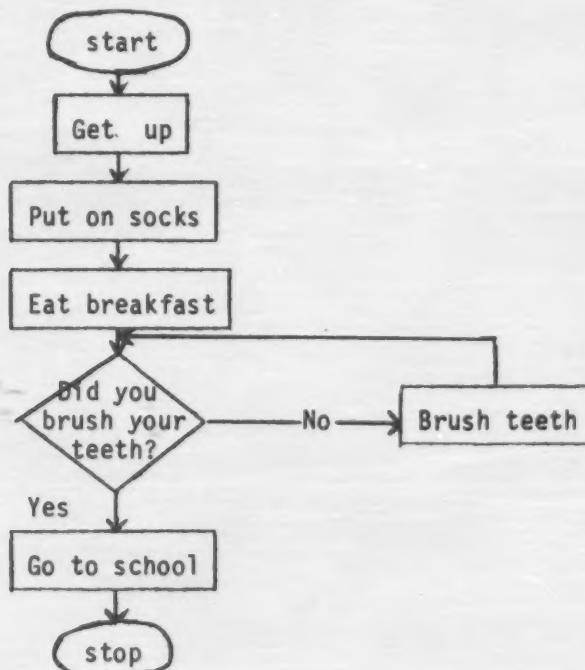
E. Introduce the decision symbol.

Asks a question



There are always at least two arrows leaving the diamond-shaped box. A yes answer will lead you in one direction. A no will lead you in another direction.

F. Put a question (decision) in the flow chart.



G. Have the students design their own flow chart about something they like to do.

Examples: go camping, bake cookies, play a game, read a book, etc.

Adapted from My Friend-the Computer by Jean Rice.

USING FLOW CHARTS TO WRITE PROGRAMS

Note: This lesson assumes students have been introduced to line numbers, the print statement, the end statement, and the mathematic operational signs (+ add, - subtract, * multiply, / divide, and exponential notation).

A. Study the sample of how to flow chart and program problems.

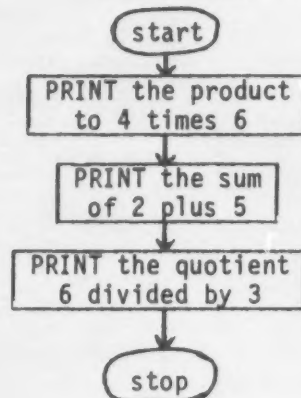
PROBLEM

4 times 6

2 plus 5

6 divided by 3

FLOW CHART



PROGRAM

10 PRINT 4*6

20 PRINT 2+5

30 PRINT 6/3

40 END

B. Draw a flow chart and write the program for the following problems.

FLOW CHART

PROGRAM

1. 4+6
9-2
8/4

2. Add 34 + 32 + 64
Subtract 336 - 292
Multiply 46 * 20

3. Multiply 15 by 23
Divide 3 by 4
Add 26 and 42

C. Variation: Given a program, have the students draw the flow chart.

MORE FLOW CHARTING

Note: This more sophisticated problem requires previous lessons on use of the REM, LET, and INPUT statements. Also, a flow chart symbol for input and output, the parallelogram.

PROBLEM

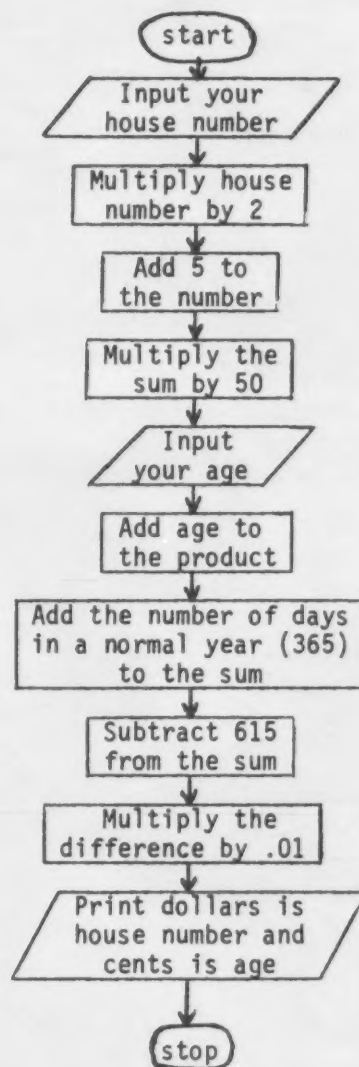
This is the process to discover a person's house number and age by telling them to:

1. Double your house number.
2. Add 5.
3. Multiply by 50.
4. Add your age.
5. Add the number of days in a normal year (365).
6. Subtract 615.
7. Put in a decimal point to show dollars and cents.
8. The dollar is your house number and the cents is your age.

PROGRAM

```
10 REM THIS PROGRAM WILL PRINT A
11 REM PERSON'S HOUSE NUMBER AND
12 REM AGE AS A DOLLAR AND CENT VALUE.
20 PRINT "WHAT IS YOUR HOUSE NUMBER";
25 INPUT A
30 LET B = A*2
40 LET C = 5+B
50 LET D = C*50
60 PRINT "WHAT IS YOUR AGE";
65 INPUT E
70 LET F = D+E
80 LET G = F+365
90 LET H = G-615
100 LET I = H*.01
110 PRINT "THE DOLLARS IS YOUR HOUSE"
111 PRINT "NUMBER AND THE CENTS IS"
112 PRINT "YOUR AGE,"; I
120 END
```

FLOW CHART



DETERMINE OUTPUT FROM INPUT

Simulate these computer runs. Show your "printout" on the screen.

1. 10 PRINT 25+35
20 PRINT 4/2, 2*3
30 PRINT 3-23; 18+2
40 PRINT 4/2, 10/2, 3-1
50 PRINT 5+4; 5-4, 5*4; 5/4
60 END
RUN

HINT: Commas placed between problems tells the computer to leave 15 character spaces between answers. A semi-colon tells the computer to print answers closer together.

2. 10 PRINT "20+6"
20 PRINT 20+6
30 PRINT
40 PRINT "BROWN, YELLOW"
50 PRINT "HELLO."
60 END
RUN

HINT: Quotation marks (" ") tell the computer to print out exactly what is between them.

3. Here is a program and "printout". Find and fix mistakes in the program so a run will produce what is shown on the screen.

10 PRINT 40+6
20 PRINT 30*4
30 PRINT
40 PRINT "50-7"
50 PRINT 121/121
60 PRINT BYE
70 END
RUN

46
34
53
1 BYE

HINT: A semi-colon at the end of one line tells the computer to print the answer to the next line beside the answer to the previous line.

MSF/1-81/ITTBPEC

ELABORATE ON PROGRAMS

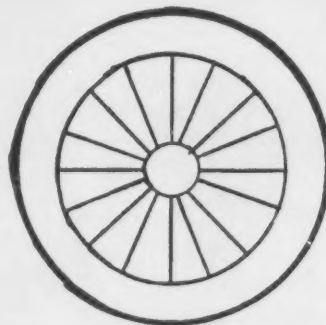
As new statements are learned the students can change the original program.

PROBLEM: A bicycle wheel is 16 inches in diameter. How far will the bike travel during one revolution of the wheel?
(Hint: Use the circumference of a wheel formula, $C = \pi D$.)

1. Using the PRINT statement only.

```
10 PRINT 3.14*16
20 END
RUN

50.24
```



2. Add the LET statement.

```
10 LET P = 3.14
20 LET D = 16
30 PRINT P*D
40 END
RUN

50.24
```

3. Add the INPUT statement.

```
10 PRINT "WHEEL DIAMETER";
20 INPUT D
30 LET P = 3.14
40 PRINT P*D
50 END
RUN

WHEEL DIAMETER? 16
50.24
```

4. Add the GOTO statement.

```
10 PRINT "WHEEL DIAMETER";
20 INPUT D
30 LET P = 3.14
40 PRINT P*D
50 GOTO 10
60 END
RUN
```

```

WHEEL DIAMETER? 16
50.24
WHEEL DIAMETER? 24
75.36
WHEEL DIAMETER? 26
81.64
WHEEL DIAMETER?

```

(Press CTRL and C together to stop execution of the program.)

5. Add the REM statement.

```

10 REM BICYCLE PROGRAM
20 PRINT "WHEEL DIAMETER";
30 INPUT D
40 LET P = 3.14
50 PRINT P*D
60 GOTO 20
70 END
RUN

```

```

WHEEL DIAMETER? 16
50.24
WHEEL DIAMETER?
(Press CTRL and C)

```

6. Final elaboration for clarity.

```

10 REM BICYCLE PROGRAM
20 PRINT "IF YOU ENTER THE DIAMETER OF A BICYCLE"
25 PRINT "WHEEL, THEN I WILL TYPE THE DISTANCE"
30 PRINT "TRAVELED IN ONE TURN OF THE WHEEL."
40 PRINT
50 PRINT "WHEEL DIAMETER";
60 INPUT D
70 LET P = 3.14
80 PRINT "DISTANCE IN ONE TURN IS"; P*D
90 GOTO 40
100 END
RUN

```

```

IF YOU ENTER THE DIAMETER OF A BICYCLE
WHEEL, THEN I WILL TYPE THE DISTANCE
TRAVELED IN ONE TURN OF THE WHEEL.

```

```

WHEEL DIAMETER? 16
DISTANCE IN ONE TURN IS 50.24
WHEEL DIAMETER? 24
DISTANCE IN ONE TURN IS 75.36
WHEEL DIAMETER? 26
DISTANCE IN ONE TURN IS 81.64
WHEEL DIAMETER?
(Press CTRL and C)

```

MORE ABOUT ELABORATION - SPACE SHIP LAUNCH

1. Original assignment.

NOTE: Parenthesis () added for clarification to audience. This information is not included in a student assignment.

Write a program that...

A. Prints out the following:

"Central Control of Spaceship Enterprise to all crew members:
All units have been checked and are ready to blast off for
Venus. All personnel to stations. Secure safety belts. Ready."

B. Uses the IF-THEN, GOTO statements and a counter to print out the countdown from 10 to 1.

```
(
100 LET X = 11
110 LET X = X-1
120 PRINT X
130 IF X = 0 THEN 150
140 GOTO 110
)
```

C. Prints "Blastoff!"

D. And prints out a picture of a rocket blasting off. (graphics)

2. Elaborated program.

Line 40	INPUT statement
Line 50	RMD (random number) library function
Lines 150-170	FOR-TO-NEXT loop with STEP
Lines 220-370	TAB(X) statement
Lines 380-400	FOR-TO-NEXT loop with numeric expression
Line 460	ON-GOTO branching statement

NUMERICAL EXPRESSIONS

Basic expressions are scanned from left to right. 

1. The computer does arithmetic in left to right order, with all multiplications (*) and/or divisions (/) performed before additions (+) and/or subtractions (-).

- A. Show the order in which the computer does the arithmetic by putting numbers 1, 2, 3, etc. in the circles above the operation symbols.

a) $\overset{\textcircled{1}}{2} + \overset{\textcircled{2}}{4} - 4$

b) $2 - \overset{\textcircled{2}}{3} * \overset{\textcircled{1}}{4}$

c) $2 * \overset{\circ}{3} / \overset{\circ}{4}$

d) $2 + \overset{\circ}{3} * \overset{\circ}{4} - \overset{\circ}{2}$

e) $2 + \overset{\circ}{3} / \overset{\circ}{4} * \overset{\circ}{2}$

f) $2 - \overset{\circ}{3} * \overset{\circ}{4} + \overset{\circ}{1} / \overset{\circ}{4}$

- B. Compute the value of the expression as a computer would solve the problem.

EXPRESSION	HOW SOLVED	ANSWER
$2 * 3 - 4$	$6 - 4$	2
$2 + 3 * 4$	$2 + 12$	14
$2 * 3 + 4 * 5$	_____	_____
$2 + 3 * 4 - 5$	_____	_____
$2 * 3 + 4 * 5 - 6 * 7$	_____	_____

2. To change the order in which operations are done, use parenthesis. Starting with the leftmost inner set of parenthesis, the computer does arithmetic within each set of parenthesis in left to right order, and then all multiplications/divisions before additions/subtractions.

- A. Show the order

a) $3 + (\overset{\circ}{3} - \overset{\circ}{4})$

b) $2 / (\overset{\circ}{3} / \overset{\circ}{4}) * (\overset{\circ}{2} + \overset{\circ}{3})$

c) $2 + (\overset{\circ}{3} + \overset{\circ}{1}) + \overset{\circ}{8}$

d) $1 + \overset{\circ}{1} / (\overset{\circ}{2} + \overset{\circ}{1} / (\overset{\circ}{3} + \overset{\circ}{1} / \overset{\circ}{4}))$

B. Compute the value.

<u>EXPRESSION</u>	<u>HOW SOLVED</u>	<u>ANSWER</u>
$(2 + 3) / (4 * 5)$	5 / 20	.25
$2 + 3 * (4 + 5)$	_____	_____
$1 / (3 + 5)$	_____	_____

C. Example of usage.

Formula for converting degrees Fahrenheit to degrees Celcius.

$$c = \frac{5}{9} (F-32)$$

$$\text{LET } C = (5/9)*(F-32)$$

3. Exponentiation ($\uparrow$), or raising to a power is performed in left to right order after parenthesis, but before all multiplications/divisions and additions/subtractions.

A. Example of usage.

Formula for interest on savings account compounded yearly.

$$A = P(1 + \frac{R}{100})^N$$

$$\text{LET } A = P*(1+R/100)\uparrow N$$

B. Sample problem.

Suppose you put \$123 into a savings account that pays 6% interest per year, compounded yearly. How much money is in the account after 2 years?

Using the LET and PRINT statements write a program to solve this problem.

C. Sample response.

```

10 LET P = 123
20 LET R = 6
30 LET N = 2
40 LET A = P*(1+R/100)\uparrow N
50 PRINT "$"; A
60 END
RUN

```

\$138.203

RESOURCES

Bob Albrecht, Basic for Home Computers: A Self-Teaching Guide, John Wiley & Sons, Inc. New York, 1978.

Gary Bitter, Basic for Beginners, McGraw Hill Book Co., New York, 1978.

Jean Rice, My Friend - the Computer, T. S. Dennison & Company, Inc., Minneapolis, Minnesota, 1976.

Jean Rice, Teacher's Guide and Activity Book to accompany "My Friend - the Computer", T. S. Dennison & Company, Inc., Minneapolis, Minnesota, 1976.

Donald Spencer, Computers for Kids, Camelot Publishing Co., Inc., Ormond Beach, Florida, 1978.

LANGUAGE - AN ALTERNATIVE APPROACH TO PROGRAMMING

Presented at:

A.S.U. Micro Computer Conference
January 16, 1981

Presented by:

Eloise C. Babcock
Teacher in Gifted Program
Navajo School
4525 N. Granite Reef Rd.
Scottsdale, Az. 85253

Outline of Teaching Sessions

I. Computer Operation Commands

Load
Return
Run
List
Save
New

II. Review Session I

Line Numbers
Rem
Print " "
Print: Print
Goto

III. Review Session II

To clear the screen
Input A
Input A
If.....then....

IV. Review Sessions II and III

Timer loop

For....Next

Tab

V. Review Sessions II, III, and IV

Loop
Gosub
Return

VI. Review Sessions II, III, IV, and V

Nested Loops
End
Stop

I. Computer Operation Commands (to operate the computer)

Load - To get the computer program off the cassette and into the computer, key in LOAD and press the return key.

Return - The computer doesn't recognize what is keyed in until the RETURN key is pressed.

Run - When you key in RUN, you tell the computer to do the program - to RUN the program.

List - When you key in LIST, you tell the computer to show you the program.

Save - When you key in SAVE, you tell the computer to put your program on a cassette tape - to SAVE your program. (Don't forget to put a cassette tape in your recorder and to press play-record buttons.)

New - When you key in NEW, you tell the computer to erase the current program and to get ready for something else - something NEW.

Cursor - The blinking spot on the monitor that tells you the computer is waiting.

To delete lines - Key in line number of unwanted line(s) and press RETURN. The line is removed.

II. Review Session I

Computer Programming in BASIC

A computer is like a calculator or a typewriter. It just sits there, waiting for someone to tell it what to do. A PROGRAM tells a computer what to do.

Line Numbers - The computer does only one command at a time. To tell it the order of operation, you use line numbers. The computer knows number order and will do your commands in order. Number by 10's - that leaves room for you to put in more lines in case you forget something or want to change something.

Rem - Remark line - The computer ignores this line. It is information for the person reading or writing the program.

Print - A PRINT without quotation marks will print a line of blanks (Looks like a skipped line).

Print "We are learning to program!" - PRINT with quotation marks tells the computer to print exactly everything that is within the quotation marks. If it's misspelled, the computer prints it misspelled.

Print:Print - Colon tells the computer that there are two separate commands on the same line. PRINT:PRINT prints two blank lines.

Goto - Tells the computer to skip to the line number mentioned after the GOTO.

Be sure to write an example on the chalkboard during your explanation.

Example:

```
10 Rem Jim Davis
20 Print "Jim is super!"
30 Print
40 Print "Jim is wonderful!"
50 Print
60 Print "Jim is super wonderful!"
70 Print:Print
80 Goto 20
```

Assignment: Students write a program using all of the commands learned today! After teacher okay, each tries own program on computer.

III. Session III Review Session II

To clear the screen and put cursor in home position:

```
Pet - 10 Print "  "
Apple - 10 Home
```

Input - as PRINT tells the computer to print the line, INPUT tells the computer a question is being asked and it waits for a response from the keyboard.

Input A - the A or any other letter tells the computer the answer will be in numbers (no letters).

Input A\$ - the A\$ or any other letter followed by a \$ tells the computer the answer will be in letters (no numbers).

If...then... - sets a condition for the computer to follow.

```
20 If A = 7 then 45 (no quotes around the number seven)
20 If B$ = "yes" then 45 (must have quotes around letters)
```

Both examples above sent the computer to line 45 if the condition is met.

Example:

```
40 Input "How old are you?"; A
50 If A < 10 then 90
60 If A > 10 then 110
70 Print "You are just my age!"
80 Goto 120
90 Print "You are younger than I am!"
100 Goto 120
110 Print "You are older than I am!"
120 Input "Do you like school?"; D$
130 If D$ = "no" then 150
140 Print "I like school, too!":Goto 160
150 Print "Too bad you don't like school!"
160 Print:Print "Time's up! Got to go now!"
```

Assignment: Students write a program using all of the commands from Sessions II and III. After teacher approval, try it on the computer.

IV. Review Sessions II and III

Timer loop - technique for making the computer pause before executing the next command.

40 FOR X = 1 to 500: Next X tells the computer to count from 1 to 500 before continuing. The higher the number, the longer the pause.

Tab - used with PRINT and functions like the tab key on a typewriter.

170 Print tab(10); "Hello!" will cause the computer to space over 10 spaces before printing HELLO!

Assignment: Students will write a program using all commands learned in Sessions II, III and IV. Insist that this program be different from the first two. After teacher approval, try it on the computer.

V. Review Sessions II, III and IV

Loop - tells the computer to repeat an operation a specific number of times.

```
180 For J = 1 to 10
190 Print tab(J);"Hello!"
200 Print:Print
210 Next J
```


Gosub - functions like a GOTO except it returns to the line number after the GOSUB command. Handy for routines that are used several times in the program.

Return - must follow the GOSUB routine. It signals the computer to return to the line number after the GOSUB.

```
240 Gosub 500
250 Print "It is a pleasure to achieve what you
    want on the computer!"
500 Print "This is our first gosub!"
510 Print "We learn by doing!"
520 Return
```

Assignment: Students will write a program using all commands learned in Sessions II, III, IV and V. Suggest that they should try to make this program as entertaining as possible. After teacher approval, try it on the computer!

VI. Review Sessions II, III, IV and V

Nested Loops - a loop within a loop.

```
120 For L = 1 to 5
130 For M = 1 to 10
140 Print tab(M); "Superb!"
150 Print
160 Next M
170 Print:Print
180 Next L
```

End - tells the computer the program is finished.

Stop - tells the computer to stop.

On C goto 240, 100, 80 - sends 1 to line 240, 2 to line 110 and 3 to line 80.

Assignment: Write a Haunted House Program
 or
 Write a Game Show Program
 or
 Write a program of your choice.

Use every command learned. It must have at least 45 lines.

VII. Review all terms and answer all questions from students.

Give students a program for them to proofread and correct all errors. (Easier on teacher's eyes if you have students use colored pencils or felt pens.)

*On the following pages is a program written to help you understand BASIC. The program is all in caps, and the explanation of what is happening is in lower case letters. It should be self explanatory.

```

10 REM ELOISE C. BABCOCK
20 REM JANUARY, 1981
30 REM HAUNTED HOUSE PROGRAM
40 REM THIS PROGRAM WAS WRITTEN AS A TEACHING TOOL - TO TEACH
    TEACHERS TO UNDERSTAND AND TEACH BASIC.

```

Rem statement - computer ignores for the convenience of the person reading or writing the program.

```

50 PRINT "♥" - clears the screen
60 PRINT
70 PRINT
80 PRINT "THE HAUNTED HOUSE!"
90 PRINT:PRINT

```

Print statements:

- Print with nothing following tells computer to print line of blanks (skips a line)
- Print with quotation marks tells computer to print exactly what appears in those quotes (same spacing, same spelling, etc.)
- Print with "♥" - code for PET to clear the screen and move the cursor (blinking square) to home position (upper left corner).
- Colon is used to separate multiple statements on same line. (lines 60 and 70 do exactly the same thing as line 90 - line 90 uses less space).

```

95 FOR X = 1 to 2000:NEXT X

```

Timer loop - used when you want computer to pause before continuing program. Nothing is printed on screen by this command.

```

100 PRINT "YOU AND YOUR FRIENDS HAVE BEEN OUT TRICK-OR-TREATING."
110 PRINT "YOU HAVE JUST COME TO THE OLD JUDSON HOUSE. EVERYONE
    SAYS IT IS HAUNTED."
120 PRINT:PRINT
130 INPUT "DO YOU WANT TO GO THERE?";A$

```

Input statements - asks questions, waits for answer from person at keyboard. Anything within quotes is printed exactly. Must be followed by semi-colon. A\$ means answer must be in letters - could be Z\$, etc. A\$ is code to regain answer later in program. If answer is numbers, the \$ is not used.

```

140 IF A$ = "YES" THEN 180

```

If...then statement - tells computer what to do under certain conditions. (line 140 says if the answer is yes then skip to line 180.)
 If answer is letters, they must be in quotes "yes".
 If answer is numbers, no quotes.
 Line 140 removes the "yes" people leaving the "no" people for line 150.

```

150 PRINT:PRINT "YOU WILL NEVER KNOW WHAT YOU MISSED!"
160 END

```

Program ends here for those who choose not to enter the house.

```

180 PRINT "YOU GO UP TO THE DOOR AND KNOCK!"
190 PRINT "APPARENTLY THE DOOR WAS NOT SHUT TIGHT BECAUSE IT
    SWINGS"
200 PRINT "OPEN AS YOU TOUCH IT!"
210 INPUT "DO YOU (1) GO IN (2) KNOCK AGAIN, OR (3) RUN?";B
220 IF B = 1 THEN 350
230 IF B = 2 THEN 300
240 PRINT:PRINT "AS YOU RUN DOWN THE DRIVE YOU HEAR STRANGE"
250 PRINT "LAUGHTER. YOU RUN FASTER AND FASTER."
260 PRINT "YOU ARE SO SHOOK THAT ALL OF YOU FORGET TO DO ANY"
270 PRINT "MORE TRICK-OR-TREATING!"
280 PRINT:PRINT "THE END!"
290 GOTO 160
300 PRINT:PRINT "AS YOU KNOCK YOU HEAR A WEAK VOICE SAY"
310 PRINT TAB(10); "COME IN, LITTLE ONES!"
320 PRINT:PRINT

```

Tab means it spaces in 10 places and then prints the statement.

```

330 INPUT "DO YOU (1) GO IN (2) RUN";C
340 ON C GOTO 350, 240

```

Tells the computer to send answer 1 to line 350 and answer 2 to line 240.

```

350 PRINT "AS YOU QUIETLY ENTER THE HOUSE, YOU ARE SURPRISED"
360 PRINT "BY HOW CLEAN AND ORDERLY IT LOOKS. THE OUTSIDE"
370 PRINT "IS A DISASTER!"
380 PRINT:PRINT "YOU ARE IN AN OLD-FASHIONED ENTRY HALL!"
390 PRINT "YOU SEE THREE CLOSED DOORS!"
410 INPUT "WHICH DOOR DO YOU TRY? (1) EAST (2) WEST (3) NORTH";D
420 PRINT "♥": PRINT:PRINT
430 ON D GOSUB 1000, 2000
440 IF D=1 or D=2 THEN 410

```

Gosub is similar to a GOTO. Goto tells the computer to skip to line number mentioned. Gosub tells the computer to skip to the gosub number, continue operations until it finds RETURN. Then it returns to the line number following the Gosub command.

```

450 REM ***NORTH DOOR ROUTINE***
460 PRINT "AS YOU APPROACH THE DOOR A VOICE SAYS:"
470 PRINT TAB(10); "TO ENTER IS EASY!"
480 PRINT TAB(12); "TO LEAVE IS HARD!"
490 PRINT:INPUT "DO YOU OPEN THE NORTH DOOR?";F$
500 IF F$ = "YES" THEN 570
510 PRINT:PRINT "YOU AND YOUR FRIENDS QUIETLY GO TO THE FRONT DOOR."

```

```

520 PRINT "IT IS CLOSED AND YOU REMEMBER YOU LEFT IT OPEN."
530 PRINT "AS YOU TRY TO OPEN IT, YOU FIND IT IS LOCKED."
540 PRINT:PRINT "AFTER TALKING IT OVER, YOU DECIDE THE ONLY THING
      LEFT"
550 PRINT "TO DO IS TO OPEN THE NORTH DOOR!"
570 PRINT:PRINT "YOUR HAND IS TREMBLING AS YOU REACH FOR THE KNOB!"
580 FOR X = 1 to 3000:NEXT X
590 PRINT:PRINT "YOU FINALLY OPEN THE DOOR!"
600 PRINT "WHAT YOU SEE INSIDE LEAVES YOU SPEECHLESS!"
610 FOR X = 1 to 9000:NEXT X
620 PRINT "☹️":PRINT:PRINT:PRINT
630 PRINT TAB(10); "S U R P R I S E ! !"
640 PRINT:PRINT:PRINT:FOR X = 1 to 2000:NEXT X
650 PRINT "INSIDE ARE ALL YOUR CLASSMATES WAITING FOR"
660 PRINT "YOU TO JOIN THEM FOR A"
670 PRINT:PRINT TAB(10); "H A L L O W E E N"
680 PRINT:PRINT:PRINT TAB(15); "P A R T Y"
690 FOR X = 1 to 1500:NEXT X
700 PRINT:PRINT:PRINT "YOU ALL SAY THIS WAS THE BEST SURPRISE"
710 PRINT "YOU EVER HAD!"
999 END

1000 REM ***EAST DOOR ROUTINE***
1010 PRINT "THE EAST DOOR IS LOCKED. YOU LISTEN";
1020 PRINT "AND HEAR NOTHING!"

```

The semi-colon tells the computer to continue printing on the same line.

```

1030 RETURN
2000 REM ***WEST DOOR ROUTINE***
2010 PRINT "THE WEST DOOR OPENS! THE ROOM IS VERY"
2020 PRINT "DARK AND FILLED WITH DUST AND COBWEBS!"
2030 PRINT "OBVIOUSLY THE ROOM IS SELDOM USED!"
2040 INPUT "DO YOU GO IN?";E$
2050 IF E$ = "NO" THEN 2090
2060 PRINT "AS YOU ENTER THE DUSTY ROOM, YOU NOTICE THAT"
2070 PRINT "ONE OF THE SHEETS COVERING THE FURNITURE IS"
2080 PRINT "MOVING SLIGHTLY. YOU ARE SCARED! YOU ALL RUN BACK TO
      THE ENTRY HALL!"
2090 RETURN

```


NAME _____ ASSIGNMENT
SESSION II

BASIC-PROGRAM EVALUATION

LINE NUMBERS _____

REM _____

PRINT " " _____

PRINT _____

PRINT:PRINT _____

GOTO _____

COMMENTS:

2.117

NAME _____ ASSIGNMENT
SESSION III

BASIC-PROGRAM EVALUATION

LINE NUMBERS _____

REM _____

PRINT " " _____

PRINT _____

PRINT:PRINT _____

GOTO _____

CLEAR SCREEN _____

INPUT A _____

INPUT A\$ _____

IF...THEN _____

COMMENTS:

NAME _____ ASSIGNMENT
SESSION IV

BASIC-PROGRAM EVALUATION

LINE NUMBERS _____

REM _____

PRINT " " _____

PRINT _____

PRINT:PRINT _____

GOTO _____

CLEAR SCREEN _____

INPUT A _____

INPUT A\$ _____

IF...THEN _____

TIMER LOOP _____

TAB _____

COMMENTS:

NAME _____ ASSIGNMENT
SESSION V

BASIC-PROGRAM EVALUATION

REM _____

PRINT " " _____

PRINT _____

PRINT:PRINT _____

GOTO _____

CLEAR SCREEN _____

INPUT A _____

INPUT A\$ _____

IF...THEN _____

TIMER LOOP _____

TAB _____

LOOP _____

GOSUB _____

RETURN _____

COMMENTS:

NAME _____

ASSIGNMENT
SESSION VI

BASIC-PROGRAM EVALUATION

REM _____

PRINT " " _____

PRINT _____

PRINT:PRINT _____

GOTO _____

CLEAR SCREEN _____

INPUT A _____

INPUT A\$ _____

IF...THEN _____

TIMER LOOP _____

TAB _____

LOOP _____

GOSUB _____

RETURN _____

NESTED LOOPS _____

END _____

STOP _____

ON C GOTO _____

COMMENTS:

NAME _____

ASSIGNMENT
DATE _____

BASIC-COMMANDS USED

REM _____

PRINT " " _____

PRINT _____

GOTO _____

INPUT A _____

INPUT A\$ _____

IF...THEN _____

ON D GOTO _____

TIMER LOOP _____

LOOP _____

NESTED LOOP _____

TAB _____

GOSUB _____

RETURN _____

COMMENTS:

TI LOGO

John D'Angelo
Texas Instruments
P.O. Box 225474, M/S370
Dallas, Texas 75230

QUESTION: What is TI LOGO?

ANSWER: TI LOGO is a high-level computer language designed for all students, including young children. It is easy to use and immediately allows students to discover the powerful attributes of the computer in an environment where the user is in control of the communication.

REMARKS: TI LOGO is the result of a collaborative effort between Dr. Seymour Papert, staff members of the Artificial Intelligence Laboratory at MIT and Texas Instruments. LOGO itself is a philosophy of education created by Seymour Papert over a twelve year period and traces many of its premises to the work of Jean Piaget and his image of the child and how children think. TI LOGO utilizes and expands upon Papert's twelve years of work with LOGO on large expensive mainframe and mini-computers and makes it available at a low cost on TI's microcomputer--the TI-99/4.

QUESTION: How is TI LOGO used?

ANSWER: TI LOGO is used as one would a pencil, to communicate with and explore the environment. The student progresses from simple sketching, through art, to mathematics, writing, and eventually, complex problem solving.

REMARKS: The objective of TI LOGO is to have control over the computer - to teach it rather than having the computer teach the user. TI LOGO will allow the student to experience computer based environments in which language, mathematics and other areas of traditional learning can be learned in a natural fashion, much as a child learns to speak. Through TI LOGO, Papert hopes to help children become articulate about and thus gain control over the learning process.

QUESTION: Where does TI LOGO fit in the school program?

ANSWER: TI LOGO initially fits as part of a school's computer literacy curriculum.

REMARKS: As computers become more pervasive in our culture during the 1980's, it is imperative that our schools provide a learning environment that will allow children to understand and appreciate what the computer can and cannot do. A second objective is to allow the students to use the computer to accomplish what they desire by using a child-oriented computer language which allows them to utilize the computer as a tool for communicating and problem solving.

TI LOGO is a child-appropriate computer language which allows a student to control the computer. Just as literacy in a language means communicating by using the ability to read and write, and literacy in mathematics means the ability to solve problems, not simply add and subtract numbers, the ability to control and program a computer is a key element in any computer literacy program implemented by a school system, and is a required skill for a person to be considered "computer literate".

TI LOGO is a computer language designed so all students, even young children, can immediately experience this control over the computer!

QUESTION: What are the benefits of TI LOGO?

ANSWER: TI LOGO assists the child in developing a process for logical and creative thinking.

REMARKS: TI LOGO permits every child to program the computer rather than the reverse as in most computer assisted instructional (CAI) applications. The use of TI LOGO by children fosters computer literacy, enriches children's logical thinking skills, and develops a healthy and positive attitude toward computers. Since TI LOGO requires little formal instruction, children can, on their own, progress in many directions and explore the world of computing. This allows them virtually unlimited creativity. Extensive field testing in the classroom has proven that students, even primary grade age children, are successful in programming TI LOGO procedures and exhibiting control over the computer.

QUESTION: What do you need to implement TI LOGO into the classroom?

ANSWER: TI LOGO can be immediately up and running on a standard TI-99/4 microcomputer equipped with a disk drive, controller and a 32K byte memory expansion.

REMARKS: Equipment and materials necessary to use TI LOGO are:

	<u>EQUIPMENT</u>	<u>MATERIALS</u>
1-TI 99/4	1-Disk Controller	1-TI LOGO Command Module
1-Monitor/T.V.	1-RF Modulator	1-LOGO Diskette
1-Disk Drive	(if TV used)	1-LOGO Guide
	1-Memory Expansion	

QUESTION: Where can I obtain more information on TI LOGO?

ANSWER: Please Write: Texas Instruments Incorporated
Corporate Engineering Center
12860 Hillcrest, Wing E, M/S 370
Dallas, Texas 75230

ATTENTION: TI LOGO

What is the Logo Language?

Logo is a high-level computer language which was designed especially for educational use. At the same time, it is a powerful computer language which is highly interactive and particularly well suited for graphics programming and list-processing. Logo's developers have been guided by the principle: "No threshold; no ceiling." More than 10 years of experience at MIT and elsewhere has demonstrated that children across the whole range of "mathematical aptitude" enjoy learning Logo. Logo experiments have involved elementary, junior high, high school students and college undergraduates. Some of the most heartening results have come from work with severely handicapped children. Papers describing previous Logo projects and concepts in more depth are available from the Logo Laboratory.

When you sit down at a computer which understands Logo and type on the keyboard you will be communicating with the Logo interpreter. This converts commands from the high-level Logo language into instructions the computer hardware can respond to directly. Commands that Logo is pre-programmed to understand are called *primitives*. New commands can be added by defining *procedures*.

Logo differs from other popular programming languages in several respects. A couple of them will be immediately apparent. In the first place, Logo is more like English, a feature which makes Logo more accessible both to children and to non-technically trained adults. Second, Logo was written to make relevant mathematical ideas as clear as possible. Here's an example. In Logo, the following procedure will cause the turtle³ to draw a circle (comments to the right of semi-colons are not part of the program; they are only explanatory):

```
TO CIRCLE                                ;Writing "TO" tells Logo that you are
                                         ;about to define a procedure.
                                         ;The name of the program is "CIRCLE".

REPEAT 360 [FORWARD 1 RIGHT 1]           ;repeat 360 times the following process:
                                         ;go forward one step and turn
                                         ;right one degree.

END                                       ;This completes the definition
```

Rather than viewing a circle in terms of coordinates $x^2 + y^2 = r^2$ this program is based on a description of a circle which is *local* and *intrinsic*. Children can discover it by reflecting on how *they themselves* walk a circle ("walk forward a little, turn a little, walk forward a little, ..."). They can use REPEAT (somelargenumber) without yet knowing the significance of 360. And they can experiment with large or small inputs to FORWARD and RIGHT. This experimenting will suggest a definition of a circle as the limit of regular polygons whose number of sides are increasing to infinity, or alternatively as a curve of constant curvature.

The idea of describing geometric figures as the tracings of a computer-controlled "turtle" leads to an approach to mathematics called "Turtle Geometry." This is a curriculum which incorporates the influence of computation in its choice of *concepts* as well as in its choice of activities. Some of the major themes illustrate the point: representation, local-global dichotomy, linearity, state and state change operators. Most important of all is the expression of mathematical ideas in terms of constructive, process-oriented formulations which can often be more assimilable, more in tune with intuitive modes of thought, than the axiomatic-

³See the description of turtle graphics, below.

Welcome to Logo

deductive formalisms in which these concepts are usually couched. This is described extensively in a forthcoming book by Hal Abelson and Andy diSessa, *Turtle Geometry: The Computer as a Medium for Exploring Mathematics* (MIT Press, Spring 1981).

Because it requires considerable computational resources, Logo could only be implemented on relatively expensive computers until recently. Now, however, a Logo implementation for an experimental version of the TI 99/4 computer exists and a Logo implementation for the Apple II will be available in the near future.

Logo miniworlds: What you can do in Logo

The world of Logo effects is divided into several miniworlds:

- turtle graphics
- sprite graphics
- list-processing
- arithmetic
- text editing
- music

Turtle graphics

Turtle graphics is Logo's line graphics miniworld. In it you command an entity called a turtle to move forward and backward and turn. The turtle draws lines. By programming in the turtle graphics miniworld you can create a wide range of often spectacular geometric designs. There have been several kinds of Logo turtles. The first were robotic turtles which moved around on wheels and drew designs on paper in response to Logo commands.⁴ Screen turtles are triangular pointers objects which move in response to turtle commands. They are able to draw as they move.

The basic turtle commands are FORWARD, BACK, RIGHT, LEFT, PENUP and PENDOWN. FORWARD and BACK take inputs which specify the number of turtle steps you want the turtle to move. RIGHT and LEFT take inputs specifying to the number of degrees you want the turtle to turn. In its PENUP state the turtle will move without leaving a trace. But when the turtle is in its PENDOWN state it will draw lines as it moves.

Sprite graphics

Within this miniworld you can program animated effects and games in color. You can use predefined shapes or design your own. The predesigned shapes are turtles, trucks, planes, balls and boxes. They can be made to appear in any of 15 colors against a background which can also be made to appear in any of these colors. Sprites are computer objects that can be bound to a shape and be made to move around the screen. Tiles are static computer objects which can be used to make designs which don't move. Logo sprite graphics presently exists only on the experimental TI 99/4 Logo machine.

There are thirty-two sprites in TI-Logo. Each sprite has a number of attributes, which can be changed by giving commands in the Logo language. For example, the following commands in sprite graphics Logo will cause sprite number 1 to look like a turtle, turn red, face due east and move with a speed of 30 (comments to the right of semi-colons are only explanatory; they are not Logo commands):

⁴Information about floor turtles can be obtained from Terrapin Inc., 678 Massachusetts Ave., Cambridge, MA. 02139.

Welcome to Logo

TELL SPRITE 1	;This command tells sprite 1 to obey ;the commands which follow.
LOOK.LIKE :TURTLE	;This command binds sprite 1's shape ;to the object which is bound to the ;the name "turtle".
SETCOLOR :RED	;This command binds sprite 1's color ;to the number which is bound to the ;name "red".
SETHEADING 90	;This command sets the direction of ;sprite 1's motion.
SETSPEED 30	;This command binds sprite 1's speed ;to 30.

A central feature of Logo sprite graphics (not implemented on the demonstration TI computers) is the availability of "when-demons." These entities enable the user to program games and animated sequences by setting up constraints having the general form WHEN (some condition obtains) DO (some action). We envision children using sprites and when-demons to explore speed-distance problems, to program games styled after popular video games, and to create computer "microworlds" for investigating physical laws.

List processing

A list in Logo is a sequence of objects. For example, [EENEY MEENEY MINEY MOE 1 2 3] is a list of seven elements. The elements of lists can themselves be lists, as well as "atomic" data objects: The list [CATS DOGS [WHITE ELEPHANTS]] contains three elements, the third of which is itself a list of two elements. Lists are extremely useful in computations which involve symbol manipulation, especially when dealing with complex, hierarchical data structures. Although Logo's list-processing capabilities are powerful enough to enable advanced users to implement sophisticated "artificial intelligence" programs, the basic operations are simple to use.

Here is a program that prints short sentences composed of a noun phrase and a verb phrase that are randomly selected from a list of possibilities.

MAKE "NOUN.PARTS [CATS DOGS BANANAS	;form a list of noun
[WHITE ELEPHANTS] ROSES]	;phrases
MAKE "VERB.PARTS [RUN BARK [SMELL GOOD]	;form a list of verb
[ARE GOOD TO EAT]]	;phrases

The following procedure uses an operation PICKRANDOM (which selects an item at random from a specified list)⁵ to choose a noun phrase and a verb phrase. These are then combined by the operation SENTENCE which puts its arguments together to form a list.

⁵PICKRANDOM is not a Logo built-in operation, but can be easily defined. Logo's "extensibility" (the fact that user-defined commands are invoked in exactly the same way as built-in commands) means that the teacher can supply special-purpose operations like PICKRANDOM which, so far as students are concerned, are just part of the language.

Welcome to Logo

```
TO BABBLE  
PRINT SENTENCE PICKRANDOM :NOUN.PARTS  
                PICKRANDOM :VERB.PARTS  
END
```

Now one can type REPEAT 10 [BABBLE] to produce:

```
CATS BARK  
ROSES RUN  
WHITE ELEPHANTS ARE GOOD TO EAT  
BANANAS SMELL GOOD
```

One can easily imagine how this program can be extended to generate complex sentences, poems, "mad libs," and so on.

Text editing

All microcomputer Logo implementations feature state-of-the-art, real-time text editing systems, available for writing procedures or for writing text which can then be printed out on hard copy devices.

Music

Logo primitives for setting pitch, durations, and voicings of microcomputer generated squarewaves will soon be implemented.

PERSONALIZING COMPUTER PROGRAMS TO YOUR OWN NEEDS

PRESENTED BY PAT KUHN

9068 North 28th Street
Phoenix, Arizona 85028

WHY PERSONALIZED PROGRAMS?

1. REINFORCES CURRENT CURRICULUM
2. DIRECT FEEDBACK TO STUDENTS
3. SAVES TIME AND DOLLARS

WHERE CAN THESE PROGRAMS BE UTILIZED?

1. DRILL AND PRACTICE ROUTINES
2. CAI PROGRAMS
3. DIAGNOSTIC TESTING
4. LEARNING GAMES

STEPS FOR PERSONALIZING PROGRAMS (ADD SPECIAL ROUTINES)

- * STUDENT NAME
- * TIMING, SPACING AND SPECIAL MESSAGES
- * POSITIVE AND NEGATIVE FEEDBACK
- * GRAPHICS
- * PROGRESS CHARTS OR SCORES

STEPS FOR PERSONALIZING PROGRAMS (DEVELOPING CURRICULUM)

- * CHOOSE SOFTWARE THAT IS EASILY ADAPTABLE TO PERSONALIZATION
- * SELECT PROGRAM MATERIAL GEARED TO STUDENT ABILITY
- * PROVIDE REMEDIAL ROUTINES

WHERE TO FIND SOFTWARE ?

- * TRADE MAGAZINES (COMPUTER)
- * COMPUTER/SCIENCE NEWSPAPERS
- * TRADE BOOKS (COMPUTER)
- * CONFERENCES

COURSEWARE DELIVERY SYSTEM

C O D E S

ACADEMIC COMPUTING ASSOCIATION
P.O. BOX 27561
PHOENIX, AZ. 85061

<u>COMPANY NAME</u>	<u>TYPE OF SOFTWARE</u>	<u>TRS-80</u>	<u>PET</u>	<u>APPLE</u>
Minnesota Software Inc. 5422 Fisher Street White Bean Lake, Minnesota 551110	Educational/Recreational Programs	X	X	X
Edu-Ware Services Inc. 22035 Burbank Blvd Suite 223 Woodland Hills, California 91367	Educational / Recreational Programs			X
Micro-Learningware P.O. Box 2134 N. Mankato, MN. 56001	Educational Elem. Jr. Hi., High School	X	X	
2.127 Milliken Publishing 1100 Research Blvd. St. Louis, Missouri 63132	Cousreware Elementary Math	X	X	X
Micromatics Inc. 29 E. 1800 So. Bountiful, Utah 84010	Elementary Basic Concepts Math & Reading Voice Syntheizer Programs	X		
Radio Shack TRS-80 Applications Software Sourcebook	Business Educational Recreational Home/Personal	X		
Cursor Magazine Box 550 Goleta, Ca. 93017	All Type of Imaginative Programs (Type Programs)		X	
273				279

COMPANY NAME	TYPE OF SOFTWARE	TRS-80	PET	APPLE
Dilethuim Press P.O. Box 92 Forest Grove, Oregon 97116	Applications, Mathematics Educational Graphics Games & Misc. Programs		X	
Basics & Beyond Inc. Pinesbridge Rd. Box 10 Anawalk, N.Y. 10501	Educational and Recreational Programs	X		
Bluebird Inc. 2267 23rd St. Wyandottle, Mi. 48192	Educational / Home / Personal Programs	X		
Educational Activities P.O. Box 87 Baldwin, N.Y. 11510	Educational Elementary Basics	X		
Instant Software Dept. RSI Peterborough, NH 03458	Educational / Recreational Home / Personal	X	X	
Program Design Inc. 111 Dar Court Greenwich, Ct. 06830	Educational / Recreational Home / Personal Programs	X	X	X
TYC Software 40 Stuyvesant Manor Geneseo, N.Y. 14454	Educational / Personal Program	X		
				281

<u>BOOKS</u>	<u>TYPES OF SOFTWARE</u>	<u>TRS-80</u>	<u>PET</u>	<u>APPLE</u>
57 Practical Programs & Games in Basic	Educational / Recreational	X		
Basic Computer Games More Basic Computer Games David H. Ahl	Educational / Recreational Home / Personal Programs	X TRS-80 Spec. Edition	X	X
Computer Games J. Victor Nahigion Wm S. Hodges	Business / Educational Home Programs	X	X	X
TRS - 80 Programs Tom Rigg & Phil Feldman	Applications, Educational Graphic, Mathematic Games and Miscs. Program	X		
32 Basic Programs for the Pet Computer Tom Rigg and Phil Feldman	Applications, Educational Graphic, Mathematic Games and Miscs, Program		X	

<u>COMPANY NAME</u>	<u>TYPE OF SOFTWARE</u>	<u>TRS-80</u>	<u>PET</u>	<u>APPLE</u>
Programs for Learning P.O. Box 954 New Milford, Cn. 06776	Educational Programs		X	
Cload Magazine Box 1267 Goleta, Ca. 93017	All Type of Imaginative Programs (Type Program)	X		
Abbout Educational Software 334 Westwood Ave. East Langwood, Ma. 01928	Educational Programs		X	
284				285

COMPUTER MAGAZINE

Compute
The Journal for Progressive Computing
The 6502 Resource Magazine
Pet-Apple - Atari
2 per copy \$15 per year

The Computing Teacher
The Computing Center
Eastern Oregon State College
La Grande OR 97850
\$10 per year

Creative Computing
Computer Applications and Software
\$2 per copy \$15 per year

Interface Age
McPheters, Wolfe and Jenes
6615 Sunset Blvd
Hollywood, California 90028
Monthly \$1.50 per copy

Personal Computing
Educational, Home and Business Applications
\$14 per year

Recreational Computing
People's Computer Co.
Box E
1263 El Camino Rd.
Menlo Park, California 94025
\$10 per year

Byte Publications Inc.
70 Main St.
Peterborough, N. H. 03458
\$1.50 per copy \$15 per year

Pursers Magazine
P.O. Box 466
El Dorado, California 95623
Quarterly \$4.00 per copy

COMPOUND WORDS

use JOV-3 tape

17 AND 92 "COMPOUND WORDS"

(DIRECTIONS)

22 "FIND A WORD AT THE TOP TO JOIN WITH"

26 "EACH WORD THAT APPEARS BELOW TO MAKE"

30 "COMPOUND WORDS. TYPE THE COMPOUND"

32 ?

34 "WORD AND PRESS RETURN."

95 "SOME BAND DRUM BAG"

(REPEATED INFO. AT TOP

97 "STORE WAY SHAKE RUN"

OF SCREEN)

99 "WATCH MOUTH MATE RISE"

DELETE LINE 101)

190 "THE COMPOUND WORD IS " ;A\$"."

(ANSWER)

600 "EAR", EARDRUM (DATA)

610 "HAND", HANDSOME

620 "MILK", MILKSHAKE

630 "HEAD", HEADBAND

640 "DRUG", DRUGSTORE

650 "SUN", SUNRISE

655 "BEAN", BEANBAG

660 "FREE", FREEWAY

670 "HOME", HOMERUN

680 "CLASS", CLASSMATE

690 "STOP", STOPWATCH

700 "LOUD", LOUDMOUTH

SYNONYMS-ANTONYMS-HOMONYMS

use JOV-3 tape

17 AND 92 "SYNONYMS ANTONYMS HOMONYMS" (TITLE)

22 "READ THE DEFINITIONS AT THE TOP OF THE" (DIRECTIONS)
26 "SCREEN. READ THE PAIRS OF WORDS BELOW."
30 "DECIDE WHETHER THEY ARE SYNONYMS,"
31 ?
32 "ANTONYMS OR HOMONYMS. TYPE YOUR ANSWER"
33 ?
36 "AND PRESS RETURN."

95 " ☐ R ☐ SYNONYMS ☐ MEAN ABOUT THE SAME THING." (REPEATED INFO.
97 " ☐ R ☐ ANTONYMS ☐ MEAN THE OPPOSITE." AT TOP OF SCREEN)
99 " ☐ R ☐ HOMONYMS ☐ SOUND THE SAME BUT HAVE"
100 TAB (9) "DIFFERENT SPELLINGS"
101 TAB (9) "AND MEANINGS."

190 Q\$; " ARE " ;A\$". (ANSWER)
600 "OVER - UNDER", ANTONYMS
610 "SEA - SEE", HOMONYMS (DATA)
620 "OLD - NEW", ANTONYMS
630 "LIFT - RAISE", SYNONYMS
640 "SENSE - CENTS", HOMONYMS
650 "FUNNY - COMICAL", SYNONYMS
655 "WET - DRY", ANTONYMS
660 "HAPPY - JOYFUL", SYNONYMS
670 "KNOT - NOT", HOMONYMS
680 "NEVER - ALWAYS", ANTONYMS
690 "PROFIT - LOSS", ANTONYMS
700 "REIGN - RAIN", HOMONYMS

FACT OR OPINION

use JOU-3 tape

17 AND 92 ""FACT OR OPINION" (TITLE)

22 "STUDY THE MEANING OF ☐ FACT ☐ AND ☐ OPINION ☐

26 "AT THE TOP OF THE SCREEN. READ THE"

30 "SENTENCE AND DECIDE WHETHER IT IS A" (DIRECTIONS)

31 ?

32 "FACT OR AN OPINION. TYPE YOUR"

33 ?

34 "ANSWER AND PRESS RETURN."

95 "A ☐ FACT ☐ IS SOMETHING THAT CAN BE PROVEN" (REPEATED INFO.

97 TAB (9) "TO BE TRUE." AT TOP OF SCREEN)

99 "AN ☐ OPINION ☐ IS WHAT SOMEONE THINKS."

101 TAB (13) "IT MAY OR MAY NOT BE TRUE."

190 Q\$ (ANSWER)

191 TAB (13) A\$

600 "CALIFORNIA IS THE BEST PLACE TO LIVE.", OPINION (DATA)

610 "APPLES ARE MORE TASTY THAN ORANGES.", OPINION

620 "THERE ARE 12 MONTHS IN A YEAR.", FACT

630 "IT'S ALWAYS WINDY ON TUESDAYS.", OPINION

640 "GIRLS, NOT BOYS, SHOULD DO THE DISHES.", OPINION

650 "MOST CATS CAN CLIMB TREES.", FACT

655 "TELEVISION IS NOT GOOD FOR CHILDREN.", OPINION

660 "A CIRCLE IS ROUND.", FACT

670 "THE EARTH IS A PLANET.", FACT

680 "A DIAMOND IS A PRECIOUS STONE.", FACT

690 "SOCCER IS A GREAT SPORT.", OPINION

700 "SPRING COMES AFTER WINTER.", FACT

QUESTION WORDS

use JOV-3 tape

17 AND 92 "QUESTION WORDS"

22 "READ THE QUESTION AT THE TOP OF THE" (DIRECTIONS)

26 "SCREEN. READ THE SENTENCE. DECIDE"

30 "WHICH QUESTION WORD THE LIGHTED PHRASE"

32 ?

34 "ANSWERS. TYPE THE QUESTION WORD AND"

35 ?

36 "PRESS RETURN."

95 "WHO WHAT WHEN" (REPEATED INFO. AT TOP OF SCREEN)

97 "WHERE WHY HOW" (DELETE LINES 98, 99, 101)

190 Q\$; " THE PHRASE ANSWERS THE QUESTION " ;A\$". (ANSWER)

600 "SUE LIVES ☐ FAR FROM TOWN ☐ ,", WHERE

610 "I WILL MEET YOU ☐ AFTER SCHOOL ☐ .", WHEN (DATA)

620 "TELL IT TO ☐ TED AND JOSE ☐ .", WHO

630 "HE MISSED THE TRAIN ☐ BECAUSE OF YOU ☐ .", WHY

640 "LET'S DANCE ☐ AS WELL AS WE CAN ☐ .", HOW

650 "IT WILL END ☐ IN A LITTLE WHILE ☐ .", WHEN

655 " ☐ OUR UNCLE AND AUNT ☐ WILL BE WITH US.", WHO

660 "LISA FEELS ☐ SICK TO HER STOMACH ☐ .", HOW

670 "WE WILL DRIVE ☐ PAST SHOPPING CENTERS ☐ .", WHERE

680 " ☐ YOUR GLASSES ☐ LOOK GOOD ON YOU.", WHAT

690 "THEY LAUGHED ☐ BECAUSE IT WAS SO FUNNY ☐ .", WHY

700 "PUT ☐ THE BOOK ☐ ON THE SHELF.", WHAT

GUIDE WORDS

use JOV-3 tape

17 AND 92 "GUIDE WORDS" (TITLE)

22 "STUDY THE GUIDE WORDS FROM THE" (DIRECTIONS)

26 "DICTIONARY PAGES. DECIDE WHICH PAGE"

30 "EACH WORD WOULD BE ON. TYPE THE NUMBER"

32 ?

34 "OF THE PAGE AND PRESS RETURN."

95 " 1 LEAF - LEASING 2 LEAST - LEG" (REPEATED INFO. AT
TOP OF SCREEN)

(DELETE LINES 96, 97, 98, 99,

101)

190 Q\$; " WOULD BE ON PAGE " ;A\$ "?" (ANSWER)

600 "LEAK", 1

610 "LEAVE", 2

620 "LED", 2 (DATA)

630 "LEASH", 1

640 "LEARN", 1

650 "LEAN", 1

655 "LEDGE", 2

660 "LEEK", 2

670 "LEASE", 1

680 "LEFT", 2

690 "LEAP", 1

700 "LEAF", 1

292

ABBREVIATIONS

use JOV-3 tape

- 17 AND 92 "ABBREVIATIONS" (TITLE)
- 22 "READ EACH WORD. FIND ITS ABBREVIATION" (DIRECTIONS)
- 26 "AT THE TOP. TYPE IT IN AND PRESS RETURN." (DELETE LINES 28, 30)
- 95 "MRS. LB. ETC. U.S.A." (REPEATED INFO. AT TOP
- 97 "P.M. AZ. A.M. MS. OF SCREEN)
- 99 "DR. P. MR. NO. (DELETE LINE 101)
- 190 "THE ABBREVIATION FOR " ;Q\$; "IS " ;A\$"." (ANSWER)
- 600 "ARIZONA", AZ. (DATA)
- 610 "AFTER 12 O'CLOCK MIDNIGHT", A.M.
- 620 "A MAN", MR.
- 630 "NUMBER", NO.
- 640 "UNITED STATES OF AMERICA", U.S.A.
- 650 "AFTER 12 O'CLOCK NOON", P.M.
- 655 "A WOMAN WHO MAY OR MAY NOT BE MARRIED", MS.
- 660 "DOCTOR", DR.
- 670 "POUND", LB.
- 680 "PAGE", P."
- 690 "AND SO ON", ETC.
- 700 "A MARRIED WOMAN", MRS.

PREFIXES

use JOV-3 tape

22 "STUDY THE PREFIXES AND THEIR MEANINGS"

23 ?

24 "AT THE TOP OF THE SCREEN. READ THE"

(DIRECTIONS)

25 ?

26 "PHRASE. ADD A PREFIX TO THE BOXED"

27 ?

28 "WORD TO MATCH THE MEANING OF THE"

29 ?

30 "PHRASE. TYPE THE WORD AND PRESS RETURN." 17 AND 92 "PREFIXES"
(TITLE)

95 " UN (NOT) SEMI (HALF OR PARTLY)"

97 " BI (TWO) RE (AGAIN)"

99 " PRE (BEFORE) IM OR IN (NOT)"

(REPEATED INFO. AT TOP
OF SCREEN)

Q\$; " MEANS " ;A\$ "."

600 "NOT ☐ POSSIBLE ☐ ", IMPOSSIBLE

(ANSWER)

610 " ☐ DO ☐ AGAIN", REDO

620 "NOT ☐ POPULAR ☐ ", UNPOPULAR

630 "HALF A ☐ CIRCLE ☐ ", SEMICIRCLE

640 "A ☐ CYCLE ☐ WITH TWO WHEELS", BICYCLE

650 "TO ☐ VIEW ☐ AHEAD OF TIME", PREVIEW

655 "PARTLY ☐ PRIVATE ☐ ", SEMIPRIVATE

(DATA)

660 " ☐ LIVE ☐ AGAIN", RELIVE

670 "NOT ☐ DIRECT ☐ ", INDIRECT

680 "A ☐ PLANE ☐ WITH TWO WINGS ON EACH SIDE", BIPLANE

690 "NOT ☐ KNOWN ☐ ", UNKNOWN

700 "TO ☐ PLAN ☐ AHEAD OF TIME", PREPLAN

2.138

CONTRACTIONS

use JOV, tape

17 AND 92 "CONTRACTIONS" (TITLE)

22 "READ THE CONTRACTIONS AT THE TOP OF THE" (DIRECTIONS)

26 "SCREEN. READ THE WORDS BELOW. TYPE IN"

30 "THE CONTRACTION FOR THE WORDS YOU READ"

31 ?

32 "AND PRESS RETURN."

95 "HE'LL WON'T IT'S DON'T (REPEATED INFO. AT TOP

97 "CAN'T WE'VE I'M SHE'LL OF SCREEN)

99 "I'D THEY'RE HAVEN'T SHOULDN'T (DELETE LINE 101)

190 A\$; " IS THE CONTRACTION FOR " ;Q\$"." (ANSWER)

600 " THEY ARE", THEY'RE (DATA)

610 "IT IS", IT'S

620 "SHE WILL", SHE'LL

630 "CAN NOT", CAN'T

640 "I AM", I'M

650 "DO NOT", DON'T

655 "HE WILL", HE'LL

660 "WE HAVE", WE'VE

670 "HAVE NOT", HAVEN'T

680 "I WOULD", I'D

690 "SHOULD NOT", SHOULDN'T

700 "WILL NOT", WON'T

PLURALS

use JOU-3 tape

17 AND 92 "PLURALS" (TITLE)

22 "READ THE RULES FOR MAKING WORDS PLURAL" (DIRECTIONS)

26 "(MEAN MORE THAN ONE). READ EACH WORD"

30 "TYPE ITS PLURAL FORM AND PRESS RETURN."

95 "MAKE MOST WORDS PLURAL BY ADDING -S."

96 " _____ " (40 SPACES) (USE \$ KEY
AND PRESS SHIFT)

97 "MAKE WORDS ENDING IN S,X,Z,SH,CH OR SS
PLURAL BY ADDING -ES."

98 " _____ " (SAME AS LINE 96)

99 "MAKE WORDS ENDING IN A CONSONANT PLUS"

100 "'Y' PLURAL BY CHANGING THE 'Y' TO 'I'"

(REPEATED INFO. AT
TOP OF SCREEN)

101 "AND ADDING -ES."

190 Q\$; " IN THE PLURAL IS SPELLED " ;A\$;" (ANSWER)

600 "BABY", BABIES (DATA)

610 "GRADE", GRADES

620 "FOX", FOXES

630 "CLOWN", CLOWNS

640 "HOBBY", HOBBIES

650 "BOSS", BOSSES

655 "CRACK", CRACKS

660 "LIBRARY", LIBRARIES

670 "BEACH", BEACHES

680 "NEST", NESTS

690 "FLASH", FLASHES

700 "JURY", JURIES

PRONOUNS

use JUV-3 tape

22 "READ THE SENTENCE. READ IT AGAIN"

23 ?

(DIRECTIONS)

24 "CHANGING THE BOXED WORD(S) FOR ONE OF"

25 ?

26 "THE PRONOUNS ABOVE. TYPE THE PRONOUN"

28 ?

30 "YOU CHOSE AND PRESS RETURN."

17 AND 92 "PRONOUNS" (TITLE)

95 "IT SHE HE THEY HER" (REPEATED INFO. TOP OF SCREEN)

97 "US HIM WE YOU THEM" (DELETE LINES 100,101,102)

99 " _____ " (40 spaces)

190 Q\$; " THE CORRECT PRONOUN IS " ;A\$ ". " (ANSWER)

600 " ☐ JUDY ☐ WENT TO THE STORE.", SHE (DATA)

610 "KYLE RODE ☐ HIS BICYCLE ☐ TO SCHOOL.", IT

620 "THE WIND KNOCKED ☐ TARA AND ME ☐ OVER.", US

630 "THE PRIZE WAS GIVEN TO ☐ JESSE ☐ .", HIM

640 " ☐ THE CATS ☐ DARTED THROUGH THE BUSHES.", THEY

650 " ☐ MY SISTER AND I ☐ LOVE TO SING TOGETHER.", WE

655 " ☐ GRANDPA ☐ , PLEASE COME HERE.", YOU

660 "OPEN THE DOOR FOR ☐ MRS. BATES ☐ .", HER

670 "SHOULD WE BUY ☐ THOSE MARBLES ☐ TODAY?", THEM

680 " ☐ GREG ☐ IS A GREAT ROLLER SKATER.", HE

690 "DO YOU ENJOY WATCHING ☐ HORROR MOVIES ☐ ?, THEM

700 " ☐ FRIDAY ☐ IS MY FAVORITE DAY OF THE WEEK.", IT

2,141

PIMA COMMUNITY COLLEGE
CENTER FOR THE INSTRUCTIONAL USE OF MICROCOMPUTERS

Daniel Davidson, CAUSE Project Director
2202 West Anklam Road
Tucson, AZ 85709
(602) 884-6000

Through funding provided in part by a grant from the National Science Foundation (CAUSE 790442), the Pima Community College district has established a central facility to support educational microcomputer use. This project institutes a "library" of micro-computers, peripherals and software to support a broad range of instructional use. Training, release time and technical assistance are available for project participants.

The project is divided into three phases: EDUCATION, in which training will be provided for faculty and students; EXPERIMENTATION, in which instructional use of the devices will be made on a trial basis over a wide range of disciplines; and IMPLEMENTATION, in which some areas will completely integrate micro-computers into their instructional process.

Faculty indicate interest by applying through Pima's in-house grants procedure. Any project in Math/Science/Social Science that gets students using microcomputers is eligible. The emphasis however, is on using the computer in some way to help students learn something. (EDUCATIONAL uses such as computerized record keeping are nice, but secondary to the purpose of this project.)

Possible uses:

1. Simulations and Models - The computer is programmed to behave like some other system. The student interacts with the simulation to gain insight into the workings of the system. A separate printed manual introduces the simulation, asks questions and guides the student through the program.

This uses the full intelligence of the computer and, if well done, holds the interest of the student. This technique is particularly useful if the process simulated is too expensive, too time consuming or too dangerous to do any other way. For example, a genetics breeding experiment which would take months or a nuclear process that would be very dangerous could be done this way.

2. Computer Assisted Instruction - The computer acts as a tutor, leading the student through some set of instruction. The branching ability of the computer is used to pick different responses according to the student's input. The PILOT language is simple to use and ideal for this use. This computer can also be interfaced to other A-V equipment.

3. Drill and Practice - The computer has infinite patience. It can randomly choose problems and examples and give immediate feedback.

4. Games - The computer can act as player or referee. To win, the student must know the math tables, chemical formulas, sentence construction or whatever subject matter the "regular" instruction includes. This is a useful way of instruction since the object for the student is not to learn the subject matter, but to win the game.

5. Connection to Other Equipment - The computer can control or "read" information from other devices. These could range from lab equipment to video tape recorders. Often when long series of calculations must be performed on experimental data, the point of the experiment is lost. A computer can log data, analyze it and present it in some easily understandable format.

During the past summer, approximately 25 Pima faculty began projects resulting in the instructional materials indicated below. Additionally in order to facilitate software development, several "utility" routine allowing more flexible use of the APPLE's graphics capabilities were devised.

Because of their high resolution color graphics, rugged construction and available software, APPLE II microcomputers were chosen for the direct instructional uses. All of the programs listed below are written in Applesoft BASIC.

1. ASTRONOMY * "Sunrise/Sunset"- An animated simulation of the passage of the sun through the celestial sphere. Under development are programs on planetary motion and the constellations.
2. CHEMISTRY * "Precip" - A game using chemistry precipitation rules ; "Nomenclature" - A tutorial on chemical nomenclature; "General Chemistry Drill and Practice Problems"; "Lab Safety Quiz".
3. COMPUTER SCIENCE * "Computer Components"; "Computer Operations" - an animated simulation of computer operations.

4. ECONOMICS * "Dynasty" - An improved version of the simulation by Weyman Fong.
5. GENETICS * "Genetics Terms"; "Making Gametes"; "Crosses Involving One Pair of Genes"; "Crosses Involving Two Pairs of Genes" - A set of tutorial programs on elementary genetics.
6. MATHEMATICS * "Graphing Linear Equations"; "Introduction to Matrices"; "Basic Math Programs" - Tutorials making full use of APPLE graphics.
7. PSYCHOLOGY * "Baby" - A simulation allowing students to use behavior modification techniques to change the behavior of a baby."
8. PHYSICS * "Problems Tutor" - A set of programs guiding students through verbal physics problems; "Vector" - A set of graphics programs illustrating vector operations most useful in elementary physics; "Physics Lab" - A set of simulations making heavy use of graphics. Topics included are Newton's Laws, Momentum Conservation, Wave addition, SHM vs. Rotational Motion, Optics, and the Kinetic Theory of Gasses.
9. UTILITIES (Programs to make programming easier) * "Skywriter" - A printing system allowing mixed text and graphics, varied character fonts super and sub-scripting, and true scientific notation; "Paddle Shapes" - A program allowing creation of shape tables using APPLE paddles; "Tablet shapes" - A program allowing the creation of shape tables using APPLE graphics tablet; "Shape Table Editor" - A program allowing modification of shape tables; "Weird Stuff" - A collection of graphics routines allowing "interesting" manipulation of the text and graphics screens.

10. WRITING * "Outline" - A program guiding students through the procedure of writing logical outlines.

Other projects not mentioned above include development of a computer data base and information gathering system to support Pima's Anthropology Technician Program and a system for psychology students to learn about the way they perceive imagery.

USING A MICROPROCESSOR IN CHEMISTRY AND PHYSICS

By: John Friend, Coconino High School, AZ.
3338 North 4th Street
Flagstaff, Arizona 86001

Introduction - What do you do after you learn to turn the microcomputer on and off?

- I. Sources - See attached list and suggestions.
- II. Uses in Science
 - A. Lab programs to perform some or all of the following functions:
 - 1. Generating answer from a lab problem.
 - 2. Check student calculations.
 - 3. Strengthen use of scientific notation and exponential numbers.
 - 4. Partial tutorial - rephrasing questions and/or simplifying question.
 - 5. Print-out abilities to generate reports of student data:
 - a. To generate a model,
 - b. Helping students organize their reports,
 - c. Computer checking and grading reports.
 - B. Simulations or demonstrations that will help in the following ways:
 - 1. Use of animation to simplify conceptual relationship.
 - 2. Present data and examples of classic experiments.
 - 3. Reduce cost by simulating expensive equipment.
 - C. Tutorial - The use of programs for tutorial means only is questionable.
 - 1. These types of programs can be dependent on curriculum use (descriptive or process).
 - 2. Tutorial aspects can be built into the above uses.
 - 3. Drill and practice types of programs: if demand necessary, can include a tutorial aspect.

Conceptual reading or content reading is a weak point of most all students and reading a screen can become old too. However, unlike a book, the computer's unique capabilities can be far more interesting or rewarding than the printed word and will hold their interest longer.

Source List

This is a list of some of the software companies and sources dealing with educational programs. This is not a recommendation, but a starter list:

Apple

Compress, Inc.
P. O. Box 102
Wentworth, N.H. 03282

Creative Discount Software
256 S. Robertson, Suite 2156
Beverly Hills, CA. 90211

Educational Activities
1937 Grand Ave.
Baldwin, N.Y. 11510

J & S Software
140 Reid Ave.
Fort Washington, N.Y. 11050

School District of Philadelphia
Henry R. Altschuler, CIA
Instructional Systems
5th & Luzerne St, 5th Floor
Philadelphia, PA. 19140

Micro-Educational
P. O. Box 280
Nelson Bar NSW 2315
Australia

Programs for Learning
P. O. Box 954
New Milford, CT. 06776

Dealing in more than one micro-system:

Conduit
P. O. Box 389
Iowa City, IA. 52244

Creative Computing
P. O. Box 789-M
Morristown, N.J. 07960

Mentor Software, Inc.
Box 791
Anoka, MN. 55303

TRS-80

Advanced Graphics
46 Kinsley Street
Nashua, N.H. 03060

Curriculum Applications
16 Plymouth St.
Arlington, MA 02174

Daniel Bishop
Box 125
Branson, MO. 65616

Dyna-Comp
P. O. Box 162
Webster, N.Y. 14580

Educational Software and Design
P. O. Box 2801
Flagstaff, AZ. 86003

Micro-Computer Software Systems
143 S. Cedros, Suite V101
Solana Beach, CA. 92075

Micro Physics Programs
2048 Ford St.
Brooklyn, N.Y. 11228

attachment

List of Publications

Magazines: Apple, TRS-80, etc.

Conduit
P. O. Box 389
Iowa City, IA. 52244

New Educational Publication:

Educational Electronics, Inc.
One Lincoln Plaza
New York, N.Y. 10023

Recommended for general sources and program information:

Creative Computing	Rates: \$20/year
P. O. Box 789-M	\$37/2 years
Morristown, N.J. 07960	\$53/3 years

Purser's Magazine	Rates: \$ 4/issue
P. O. Box 466	\$12/year
El Dorado, CA. 95623	

Softside Subscription Svs. Dept./Magazine Only - \$18/year	(Must specify:
P. O. Box 68	Magazine/Cassette - \$75/year
Milford, N.H. 03055	Magazine/Dis. - \$125/year
	Apple, TRS-80, or Atari)

TRS-80 users only:

80 Microcomputing
Subscription Department
Farmingdale, N.Y. 11737

80 U.S. Journal
3838 S. Warner St.
Tacoma, WA. 98409

Cload
P. O. Box 1267
Goleta, CA. 83017

Apple and other users only - not familiar with specific magazines.

Publishers such as Random House are getting into the educational software business. Your district may be receiving a catalog shortly.

COMMENTS/SUGGESTIONS

Other sources for educational programs:

1. District could set aside time or pay to have their staff members develop software and share with other districts.
2. Write you own programs.
3. Co-author:
 - a. Design your programs and have knowledgable students write them.
 - b. Cooperate with others and develop teams (designers and programs).
4. Give your ideas away and hope someone else will develop the programs.

Don't let your good ideas to to waste. There are many educational software companies that are willing to give you a fair compensation or royalty for your work. I feel that the good software will come from the trenches. Even though most teachers have more than enough to do, I feel the following recommendations are valid:

1. State universities and/or colleges should develop courses for educators ranging from general use and understanding of the microcomputers to quality programming designed for the unique features of the microcomputer. The present system or classes being offered in many universities is not enough or adequate for educators' needs.
2. Write the publishers of the text and manuals you are using and ask them to develop or recommend a source for software to use with their materials.

PEGGY COMER
SUNBURST SCHOOL, CHAPARRAL SCHOOL
14218 N. 47TH AVENUE
GLENDALE, ARIZONA 85306

BLOOM'S TAXONOMY WAS FIRST CONCEIVED IN 1948.
THROUGH THESE CATEGORIES WE ARE ABLE TO IDENTIFY VARIOUS
TYPES OF QUESTIONS. THE LEVELS ARE:

KNOWLEDGE: RECALL OF INFORMATION

COMPREHENSION: UNDERSTANDING INFORMATION

APPLICATION: USING INFORMATION

ANALYSIS: TAKING APART INFORMATION

SYNTHESIS: WORKING WITH PARTS OF INFORMATION AND
COMBINING THEM IN DIFFERENT WAYS

EVALUATION: JUDGING THE VALUE OF INFORMATION.

> SOMETIMES
ARE
COMBINED

THIS TAXONOMY CAN BE EFFECTIVELY UTILIZED IN TEACHING
A COURSE ON MICROCOMPUTERS. IT IS VERY USEFUL AS A TOOL
TO GET THE STUDENTS INVOLVED IN USING THE COMPUTER FOR
SOMETHING OTHER THAN GAME PLAYING AND PROGRAMMING. WHILE
THE FORMER ACTIVITIES ARE AN IMPORTANT PART OF COMPUTER
EDUCATION, I FEEL, THERE SHOULD BE MORE INVOLVED.

THEREFORE, I DESIGNED A LEARNING CENTER WHICH USES
THESE DIFFERENT LEVELS OF QUESTIONING AND INVOLVES
TOPICS SUCH AS ROBOTS, HISTORY, CAREERS, PRIVACY, FUTURE,
APPLICATIONS, TECHNOLOGY AND ACTUAL USE OF THE COMPUTER.
EXAMPLES OF TASK CARDS FOLLOW:

ROBOTS -- SYNTHESIS: COMPOSE A NEWS ARTICLE ON THE USE
OF ROBOTS ON THE ASSEMBLY LINE.

HISTORY -- RECALL AND APPLICATION: LOCATE SEVERAL BOOKS
IN THE LIBRARY (SCHOOL OR PUBLIC) ON THE HISTORY

OF COMPUTERS. ORGANIZE A SCRAPBOOK GIVING A TIME-
LINE ACCOUNT OF COMPUTER DEVELOPMENT.

CAREERS -- APPLICATION: COLLECT NEWSPAPER ADVERTISEMENTS
FOR JOBS DEALING WITH COMPUTERS. CLASSIFY ACCORDING
TO JOB, SALARY, EXPERIENCE REQUIRED, EDUCATION.

PRIVACY -- RECALL AND EVALUATION: EXPLAIN HOW COMPUTERS
INVADE AN INDIVIDUAL'S PRIVACY. PREDICT THE RESULTS
OF A SOCIETY WHERE THERE ARE NO SECRETS AND NO PRI-
VACY.

FUTURE -- SYNTHESIS: ORIGINATE A COMPUTERIZED SYSTEM TO
REPLACE TEACHERS.

APPLICATIONS -- ANALYSIS: COMPARE THE GOOD AND BAD
EFFECTS OF COMPUTERS ON ORGANIZATIONS.

TECHNOLOGY -- ANALYSIS: COMPARE A COMPUTER TO A HUMAN
BEING. LIST THE ADVANTAGES AND DISADVANTAGES OF
EACH.

ACTUAL USE -- SYNTHESIS: DESIGN A CAI SYSTEM THAT WOULD
HAVE BEEN USEFUL IN TEACHING THIS CLASS.

PRIVACY AND COMPUTERS

Recall/Evaluation

Explain how computers invade an individual's privacy. Predict the results of a society where there are no secrets and no privacy.

Application

Sweden has had privacy legislation since 1973. Collect information on privacy legislation and present your findings to the class (orally and written).

Application

Keep a diary for 1 month. Record everything. You don't need to share your "secrets", but how much of what you do in 1 month would you like to keep to yourself without everyone knowing?

Evaluation

Consider whether businesses need privacy.

Analysis

Analyze the difference between privacy and secrecy.

Evaluation

Debate the following:
"The individual has a right to privacy. Society has a right to be informed."

FUTURE

Recall/Application

Locate 5 science fiction books which revolve around computers. Choose 1 and report on it.

Synthesis

Design a magazine called "Computers in Medicine" which will be published in the year 2401.

Synthesis

Add to the ways in which computers could be used in the medical profession in the future.

Synthesis

Originate a computerized system to replace teachers.

Evaluation

Write an editorial on computers controlling our lives.

Application/Analysis

Choose a current prediction of the future and analyze it.

Synthesis

Create a cartoon for the year 3084 when everything is controlled by a computer.

Synthesis

Predict what life controlled by a computer would be like.

Evaluation

Justify a cashless society.

CAREERS

Recall

Identify various tasks an experienced programmer might be expected to do.

Recall

Make a chart listing offerings at computer schools in the Phoenix area.

Analysis

Compare the responsibility of a computer specialist and a nuclear bomb scientist.

Synthesis

What if you were a newspaper editor... how would you use a computer to help in running the newspaper?

Recall

Locate some schools in the Phoenix area which offer training in a computer-related field.

Application

Collect newspaper advertisements for jobs dealing with computers. Classify according to job, salary, experience required, education.

Synthesis

Imagine you are applying for a job as a data processing manager in a medium sized computer installation. Write a letter to the company giving a resume of your previous employment, education, and qualifications.

Evaluation

Debate whether computers have created or eliminated jobs.

HISTORY

Recall and Application

Locate several books in the library (school or public) on the history of computers. Organize a scrapbook giving a time-line account of computer development.

Application

Construct a time-line on the development of computers from early man to the present.

Analysis

Compare the development of data processing equipment and communication facilities. List the dates, people, and equipment involved.

Analysis

Compare a calculator and a computer.

Recall

Demonstrate an abacus to the class.

Application and Synthesis

Choose the person who made the greatest contributions towards the development of the computer. Give reasons for your choice.

Analysis

Make a graph or chart comparing the first generation computers with present day computers. (Cost, processing time, size, etc.)

Analysis

Compare a computer of the 1940's to one of the present day.

ROBOTS

Recall

Locate a magazine or newspaper article on robots.

Analysis

Compare R2D2 with C3PO.

Synthesis

Create a story in which the hero is a robot.

Evaluation

Recommend Asimov's book, I Robot to a friend.

Application/Analysis

Sketch 10 robots which have been used in movies. Give a brief personality analysis of 2 of them.

Synthesis

Design a robot which does a specific job.

Synthesis

Compose a news article on using robots on the assembly line.

COMPUTER APPLICATION

Recall

List any uses of computers that personally affect you or your family.

Application

Choose a specific business and find out how and why it uses computers.

Application

Research ways in which businesses use computers. Report on one application to the class.

Analysis

Compare contributions which the telephone, airplane, computer and car have made to mankind.

Analysis

Survey local TV stations to see how computers are used and compare the results.

Analysis

Prepare a questionnaire to send to several businesses (airlines, banks, small business, etc.) which would find out how they use computers in their business.

Synthesis

Imagine all the ways a large shopping center could use a computer. Write a script for a news documentary.

Evaluation

Consider the question, "Can computers think?" Present your conclusions.

Recall/Analysis

List as many present-day uses for computers as possible. Separate the list into two categories:
-things that could be done before computers
-things not possible before computers.

Application

Choose 3 or 4 examples of how computers could be used to solve problems of traffic congestion.

Application

Using a newspaper, make a scrapbook of articles dealing with computers.

Analysis

Compare the good and bad effects of computers on organizations.

Analysis

Survey at least 30 of your classmates to find out how they feel about computers.

Analysis

Analyze the advantages and disadvantages of the big computer vs several small ones.

Synthesis

Compose a questionnaire which might be used for a computer dating service.

USING THE COMPUTER

Recall

Show someone how to get "on line" using a PET.

Recall

Use a chart or a diagram to explain how to save a new program on tape.

Synthesis

Invent a new game which would help someone learn to use a computer.

Recall

Demonstrate a program which will read 2 numbers and write them out.

Application

Compose your own message and write a program which will type out your own message exactly 10 times.

Synthesis

Design a CAI system that would have been useful in teaching this class.

TECHNOLOGY

Recall/Application

Explain the function of the main parts of a computing system and make a drawing or a model.

Recall

Explain the Hollerith code.

Recall

Show the different parts of a computer by making a diagram.

Analysis

Compare a computer to a human being. List the advantages and disadvantages of each.

Synthesis

What if the ideas and technology required to build and develop the computer had been available 100 years earlier. What would be different today? What different uses might have been developed for computers by now.

Recall

Illustrate a punched card. Label the parts.

Recall

List some advantages of using tape instead of cards to store data.

Analysis

Compare punched cards, magnetic tape, and magnetic disks with respect to the following:

1. storage capacity
2. speed of reading by computer
3. ease of human reference
4. ability to be re-used.

Evaluation

Mankind could not have gone to the moon without a computer. Discuss.

VOCABULARY LIST

- ABACUS A COUNTING DEVICE USED TO DO MATHEMATICS.
- ANALOG COMPUTER A COMPUTER WHICH COMPARES ONE THING WITH ANOTHER.
AN EXAMPLE IS A BATHROOM SCALE.
- ARITHMETIC UNIT THE PART OF A COMPUTER THAT DOES THE MATHEMATICS
OF A PROBLEM.
- BASIC AN EASY COMPUTER LANGUAGE USED BY BEGINNERS. (BEGINNERS
ALL-PURPOSE SYMBOLIC INSTRUCTIONAL CODE)
- BINARY ALL DIGITS ARE 0 OR 1 ; BASE 2
- BIT ABBREVIATION FOR BINARY DIGIT.
- BUGS ERRORS IN A COMPUTER PROGRAM.
- BYTE GROUPING OF BITS, USUALLY 8.
- CARD READER A MACHINE USED TO DETECT HOLES IN PUNCHED CARDS.
IT THEN SENDS SIGNALS TO A COMPUTER TO TRANSLATE
THE INFORMATION FROM THE HOLES.
- CATHODE RAY TUBE (CRT) A MACHINE, WITH A KEYBOARD LIKE A TYPE-
WRITER AND A TV SCREEN, WHICH CAN BE
USED AS BOTH AN INPUT AND OUTPUT DEVICE.
- CENTRAL PROCESSING UNIT THE PART OF THE COMPUTER WHERE THE
INFORMATION IS WORKED ON. PERFORMS
ARITHMETIC, LOGIC, AND CONTROL FUNCTIONS.
- CHIP ELECTRICAL PART WHICH IS USED IN 4TH GENERATION COMPUTERS.
- COBOL A COMPUTER LANGUAGE USED BY BUSINESS.
- COMPUTER A MACHINE, WITH A MEMORY LIKE A CALCULATOR, WHICH
ACCEPTS INFORMATION, WORKS ON THE INFORMATION TO
SOLVE A PROBLEM, AND PUTS OUT THE ANSWER.
- CONTROL UNIT A PART OF THE COMPUTER THAT DIRECTS THE FLOW OF
DATA THROUGH THE COMPUTER.
- CURSOR MOVABLE SPOT OF LIGHT ON CRT WHICH INDICATES WHERE THE
NEXT CHARACTER WILL BE ENTERED.
- DATA THE INFORMATION A COMPUTER NEEDS TO SOLVE A PROBLEM.
- DEBUG LOCATE AND CORRECT ALL ERRORS IN A PROGRAM.
- DIGITAL COMPUTER A COMPUTER, WITH A MEMORY, THAT WORKS BY
USING NUMBERS AND FOLLOWS A SET OF INSTRUCTIONS
CALLED A PROGRAM.
- EDVAC FIRST COMPUTER TO USE BINARY LANGUAGE, COMPLETED IN 1950.

VOCABULARY LIST CONTINUED

END THE STATEMENT IN A COMPUTER PROGRAM WHICH HAS THE HIGHEST STATEMENT NUMBER.

ENIAC FIRST COMPLETELY ELECTRONIC COMPUTER, COMPLETED IN 1946

FLOW CHART A DRAWING OR MAP THAT TELLS THE STEPS NEEDED TO SOLVE A PROBLEM. IT ALSO SHOWS THE CORRECT ORDER FOR DOING THE STEPS.

FORTRAN A COMPUTER LANGUAGE USED BY SCIENTISTS.

GENERATIONS COMPUTER GENERATIONS REFER TO GROUPS OF COMPUTERS USING CERTAIN KINDS OF PARTS. FOR EXAMPLE, FIRST GENERATION COMPUTERS USED VACUUM TUBES.

HARDWARE THE COMPUTER MACHINERY.

INPUT INFORMATION THAT GOES INTO THE COMPUTER.

INTEGRATED CIRCUITS ELECTRICAL CONNECTIONS BETWEEN POINTS WHICH CONTROL THIRD GENERATION COMPUTERS.

KEYPUNCH A MACHINE USED FOR PUNCHING HOLES IN CARDS.

LOG-IN TO SIGN IN ON A COMPUTER.

MAGNETIC CORES TINY DOUGHNUT-SHAPED METAL RINGS USED TO STORE INFORMATION INSIDE A COMPUTER.

MAGNETIC DISK A DEVICE WHICH LOOKS LIKE A PHONOGRAPH RECORD. IT CONTAINS INFORMATION WHICH CAN BE SENT TO THE COMPUTER.

MAGNETIC DRUM A ROUND DRUM-LIKE DEVICE WHICH HOLDS INFORMATION ON A MAGNETIC SURFACE.

MAGNETIC TAPE A PLASTIC RIBBON, WHICH RESEMBLES TAPE RECORDER TAPE. INFORMATION IS RECORDED ON THE TAPE IN THE FORM OF MAGNETIC SPOTS.

MARK I FIRST WORKING COMPUTER, COMPLETE IN 1944.

MEMORY THE PART OF THE COMPUTER WHERE PROGRAMS AND DATA ARE STORED.

MINICOMPUTER A SMALL COMPUTER ABOUT THE SIZE OF A SUITCASE.

NANOSECOND A BILLIONTH OF A SECOND.

ON-LINE USING A COMPUTER TERMINAL WHEN IT IS HOOKED UP TO A COMPUTER.

OUTPUT THE ANSWER TO A PROBLEM, WHICH COMES OUT OF A COMPUTER.

PRINT AN INSTRUCTION TO THE COMPUTER WHICH TELLS IT TO TYPE OUT A CERTAIN THING.

VOCABULARY LIST PAGE 3

PRINT OUT A FORM OF OUTPUT FROM A COMPUTER WHICH CONTAINS THE INFORMATION THE COMPUTER HAS JUST PROCESSED. IT IS ALSO CALLED A READ OUT.

PRINTER THE MACHINE USED TO PRINT THE OUTPUT FROM THE COMPUTER.

PROGRAM A SET OF INSTRUCTIONS WHICH TELLS THE COMPUTER WHAT TO DO.

PROGRAMMER A PERSON WHO WRITES A COMPUTER PROGRAM.

PUNCHED CARD A CARD WHICH MAY BE PUNCHED WITH HOLES THAT STAND FOR LETTERS, NUMBERS OR SYMBOLS.

SOFTWARE THE PROGRAMS THAT GO INTO THE COMPUTER.

STATEMENT A LINE OF A COMPUTER PROGRAM.

STATEMENT NUMBER OR LINE NUMBER A NUMBER AT THE BEGINNING OF A LINE IN A COMPUTER PROGRAM.

TERMINAL A DEVICE WITH A KEYBOARD LIKE A TYPEWRITER THAT CAN BE HOOKED UP TO A COMPUTER OVER TELEPHONE LINES.

TIME-SHARING A COMPUTER SYSTEM WHERE MANY PEOPLE USE THE COMPUTER AT THE SAME TIME.

TRANSISTOR AN ELECTRICAL PART WHICH CONTROLLED THE FLOW OF ELECTRICITY IN SECOND GENERATION COMPUTERS.

VACUUM TUBES AN ELECTRICAL PART WHICH WERE USED IN FIRST GENERATION COMPUTERS.

BIBLIOGRAPHY

- ALBRECHT, ROBERT. MY COMPUTER LIKES ME WHEN I SPEAK IN BASIC. MENLO PARK, CA: DYMAX, 1972. FOR BEGINNERS.
- APPLE PERSONAL COMPUTER MAGAZINE. "COMPUTERS IN EDUCATION," VOL. 1, NO. 1. COMPUTER USES IN EDUCATION.
- BALL, MARION AND SYLVIA CHASE. BE A COMPUTER LITERATE. MORRISTOWN, N.J.: CREATIVE COMPUTING PRESS, 1977. AN INTRODUCTION TO WRITING SIMPLE PROGRAMS IN BASIC.
- BALL, MARION J. WHAT IS A COMPUTER? HOUGHTON MIFFLIN Co., 1972. DESCRIBES COMPUTERS. GOOD FOR PRIMARY.
- BITTER, GARY AND WILSON GATELEY. BASIC FOR BEGINNERS. NEW YORK: MC-GRAW-HILL BOOK Co., 1970. INTRODUCTION TO BASIC.
- CORLISS, WILLIAM R. COMPUTERS. WASHINGTON, D.C.: USERDA, OFFICE OF PUBLIC AFFAIRS, 1973. HISTORY; ANALOG/DIGITAL COMPUTERS.
- DWYER, THOMAS AND MARGOT CRITCHFIELD. COMPUTER RESOURCE BOOK-ALGEBRA. BOSTON: HOUGHTON/MIFFLIN Co., 1975. INTRODUCTION TO BASIC USING ALGEBRA.
- DWYER, THOMAS AND MICHAEL KAUFMAN. A GUIDED TOUR TO COMPUTER PROGRAMMING IN BASIC. BOSTON: HOUGHTON/MIFFLIN Co., 1973. VERY GOOD FOR JUNIOR HIGH.
- NELSON, TED. COMPUTER LIB/DREAM MACHINES. SWARTHMORE, PA: TED NELSON, PUBLISHER (DISTRIBUTED BY THE DISTRIBUTORS, 702 S. MICHIGAN, SOUTH BEND, IN 46618. COVERS THOUGHTS ABOUT AND USES OF COMPUTERS.
- PEOPLE'S COMPUTER COMPANY. WHAT TO DO AFTER YOU HIT RETURN. MENLO PARK, CA: 1263 EL CAMINO REAL, BOX E. GAMES
- RICE JEAN. MY FRIEND-THE COMPUTER: TEACHER'S GUIDE AND ACTIVITY BOOK TO ACCOMPANY "MY FRIEND THE COMPUTER." MINNEAPOLIS: T.S. DENNISON & COMPANY, 1976. STUDENT BOOK AND TEACHER'S GUIDE. PROVIDES TRANSPARENCIES, OBJECTIVES AND RESOURCES.
- SPENCER, DONALD D. COMPUTERS IN SOCIETY: WHERE'S, WHYS AND HOWS OF COMPUTERS USE. ROCHELLE PARK, N.J.: HAYDEN BOOK Co., INC., 1974. DIFFERENT APPLICATIONS OF COMPUTERS.
- SQUIRE, ENID. THE COMPUTER, AN EVERYDAY MACHINE. MENLO PARK, CA.: ADDISON-WESLEY, LTD., 1972. PAST, PRESENT, FUTURE.

FILMS DEALING WITH COMPUTERS

1. BELL TELEPHONE Co.
 - A. INCREDIBLE MACHINES - 15 MIN.
SHOWS WAYS BELL TELEPHONE SCIENTISTS USE COMPUTERS IN COMMUNICATION RESEARCH.
 - B. THINKING MACHINE - 16 MIN.
COMPARES THINKING ABILITY OF HUMANS TO COMPUTERS.
2. CONTROL DATA CORP., 8100 34TH AVE. So., MINNEAPOLIS, MINN., 55420
 - A. THE MARVELOUS CALC. - 19 MIN.
TRACES HISTORY OF COMPUTER.
3. IBM
 - A. A COMPUTER GLOSSARY - 10 MIN.
ANIMATION - DEFINES SEVERAL MORE COMMONLY USED TERMS WHICH BELONG TO THE COMPUTER AND DATA PROCESSING FIELDS.
 - B. THE INFORMATION MACHINE - 10 MIN
DEVELOPMENT OF THE COMPUTER FROM PRIMITIVE MAN TO MACHINE SIMULATION.
 - C. MAN AND COMPUTER - A PERSPECTIVE - 20 MIN.
SHOWS BASIC ELEMENTS IN DATA PROCESSING SUCH AS INPUT, OUTPUT, STORAGE AND CONTROL.
4. THE RAND CORP., 1700 MAIN ST., SANTA MONICA, CA., 90406
 - A. TOMORROW'S PROGRAMMERS
COMMENTARY ON A COURSE IN COMPUTING WHICH WAS OFFERED FOR TWENTY-TWO GIFTED TWELVE YEAR OLDS AT SANTA MONICA HIGH SCHOOL.

Microcomputer-Assisted Instruction with Learning Disabled Students

Marley Watkins, Ph.D.¹

Deer Valley School District

17624 North 31 Ave.

Phoenix, Arizona 85023

A vigorous debate has erupted concerning the use of microcomputers in education. Conclusions have emerged which range from educational panacea to societal disaster. Those of us currently using, or considering the use of, microcomputers in education will contribute to this on-going debate and will play an important part in determining whether microcomputers become productive educational tools or whether they are eventually considered as just another educational fad. It will be unfortunate if we as educators serve only as 'testifiers' and not as scientific evaluators. The present paper will describe my efforts at evaluation and will further present opinions and evidence I have culled regarding educational software and hardware.

My interest in microcomputers began about three years ago when I recognized their tremendous potential in education. I naively thought that the hardware would become standardized, that experimental evidence would rapidly accumulate, that educationally and technically sound software would abound, and that funds would flow. I was wrong on all four counts! The microcomputer market

is still divided, software has improved little in quantity and not at all in quality, and experimental evidence of effectiveness is still meager. Funding is now more available but remains a major obstacle to many innovative applications. It is now apparent that this situation will not change for the better until we professionals enter the field as scientific users and evaluators. Only then will sound software be produced and used so that valid experimental evidence can accumulate.

We can structure this large endeavor by considering two factors. First, it is obvious that many types of learners exist (i.e., gifted, elementary school, handicapped, adult, etc.). Second, it is generally accepted that several microcomputer instructional 'modes' are viable (i. e., drill and practice, tutorial, simulations, problem solving, etc.). Our knowledge concerning the efficacy of each instructional mode across learner types is currently very limited. Without this knowledge, however, we are doomed to inefficient, trial-and-error applications costly in terms of money and misdirected human effort. This is not to advocate that we should not try microcomputers in education before evidence regarding their effectiveness is firm. It is an unequivocal call for all who use microcomputers to consider the above learner type x mode structure and to conduct competent program evaluations of their efforts so that others may share their experiences. It is my belief that the adoption of microcomputers in education on a massive scale will result in a debacle akin to modern math if this evaluative stance is not wide spread.

As outlined above, we can reduce the probability of ill-advised adoption of microcomputers in education by embarking on a systematic evaluative program utilizing learner type and instructional mode as factors. My own efforts have focused upon the drill and practice mode as applied to handicapped children. A literature review has been conducted and has generally been supportive of the efficacy of drill and practice. This reference list is provided immediately following the body of the text. Lacunae exist; most notably in applications and evaluations with Learning Disabled students. Consequently, my efforts have become even more narrowly focused upon microcomputer-assisted instruction with Learning Disabled children.

As defined by state and federal guidelines, Learning Disabled children exhibit a severe discrepancy between ability and academic achievement. They are identified in the public schools by multidisciplinary teams consisting of psychologists, teachers, administrators, and parents. Once identified, they are provided intensive academic assistance in special education programs to bring their academic achievement levels into line with their estimated ability level. The Learning Disabled child typically is not learning through regular teaching techniques, may often become very frustrated and discouraged because of previous failures, and may have developed a real antipathy toward academic subjects. Given these characteristics, it seems that these children would be good subjects for testing an alternative instructional delivery system such as microcomputer-assisted instruction.

Before testing the efficacy of microcomputer-assisted instruction (MCAI), it is imperative that we utilize appropriate software and

hardware. To do otherwise would be to conduct an unfair test. Until recently, much educational software was marked by poverty of content and/or by inadequate technical operationalization. It seems apparent that both components are necessary for optimum learning to occur. My training as an educational psychologist and my experience as a practicing school psychologist suggests several necessary components of sound educational drill and practice programs:

1. It must capture the learner's attention. This might be accomplished through sound, color graphics, novel displays, etc.
2. It must allow children and teachers to know exactly what is required of them. Ambiguity of presentation and content will defeat children and teachers.
3. Its technical quality must be adequate. Without this, ambiguity and other learning impediments will intrude.
4. It must individualize instruction.
5. It must personalize instruction.
6. It must incorporate performance objectives into a sequential hierarchy of skills to foster precision teaching, diagnostic-prescriptive, and teaching by objectives approaches.
7. It must maintain learner motivation even after the novelty effect dissipates. Years of psychological research have demonstrated that a structured positive reinforcement system can produce such an outcome.
8. It must provide immediate, non-threatening feedback to each learner response.

9. It must reduce teacher paperwork but must not encourage the teacher to relinquish control of the students' programs to the computer.
10. It must fit within existing service delivery systems.

Computer hardware is also a vital component to consider. It must be selected with care to maximize utility and minimize cost. Several major brands are readily available, each with strong and weak points which must be weighed by the educator in relation to his/her needs. A computer literacy program might call for different hardware than a CAI program. The educator will find that a careful analysis of educational needs will allow a more informed hardware selection.

After securing appropriate software and hardware, I and the Deer Valley School District have embarked upon a one year experimental analysis of the effectiveness of MCAI with elementary school Learning Disabled children. This was made possible by a Title IV-C grant entitled "Microcomputer-Assisted Instruction in Mathematics for Learning Disabled Students." Provision of funds by this agency is, in my opinion, very progressive and signifies a real commitment to innovative educational programs. The project is designed along sound quasi-experimental lines and includes random assignment of students to treatment groups as well as extensive pre and post-tests for statistical analysis of effects. Approximately 45 Learning Disabled students receive math CAI, another 45 receive spelling CAI, and approximately 180 others receive regular special education services without computer

supplementation. Eight special education teachers have included the microcomputers into their rooms and use them as instructional tools. Consequently, the existing service delivery system has remained essentially unchanged. Academic data will be analyzed and reported by the end of June. My observations have lead me to several conclusions: (a) motivation to achieve is being maintained, (b) teachers perceive the microcomputers as useful tools, (c) parents are aware of and impressed by the microcomputers' impact on their children, and (d) some children have made impressive academic gains. However, these observations are highly subjective and must be looked at as testimonials until supported by objective data. Surveys will be distributed in the coming months to gather more objective information concerning student, parent, and teacher opinions. When combined with academic achievement data, this information should enlighten us along several dimensions.

In accordance with my previously stated position regarding evaluation of microcomputer programs and sharing results with the scientific community, I would like to make the above mentioned evaluation results available. Consequently, an evaluation report will be sent to those who write to me at the title address and who enclose a self-addressed envelope with 30 cents postage affixed. This is necessitated by lack of dissemination funds. I encourage the reader to take advantage of this offer. I also strongly urge readers who possess evaluative data or reports on microcomputers in education to contact me so that I may assist in dissemination efforts.

Footnotes

¹ Opinions expressed in this paper are solely those of the author and are not necessarily representative of the Deer Valley School District or of ESEA Title IV-C.

CAI and CMI References

- Adkins, K. Success story: CAI in the elementary school. AEDS Proceedings, 1979.
- Atkinson, R. Computerized instruction and the learning process. American Psychologist, 1968, 23, 225-239.
- Bozeman, W. C. Computer-managed instruction: State of the art. AEDS Journal, 1979, 12, 117-137.
- Caldwell, R. M., & Rizza, P. J. A computer-based system of reading instruction for adult non-readers. AEDS Journal, 1979, 12, 155-162.
- Chiang, A., et al. Demonstration of the use of computer-assisted instruction with handicapped children. 1978 (ERIC Document Reproduction Service No. ED 166913).
- Chizman, J. F., Hiebert, L. D., & McCarney, B. J. Assessing the impact of an instructional innovation on achievement differentials: The case of computer-assisted instruction. The Journal of Economic Education, 1977, 9, 42-46.
- Diem, R. A., & Fairweather, P. G. An evaluation of a computer-assisted education system in an untraditional setting - a county jail. AEDS Journal, 1980, 13, 204-213.
- Edwards, J., Norton, S., Taylor, S., Weiss, M., & Dusseldorp, R. How effective is CAI? A review of the research. Educational Leadership,

1975, 147-153.

Fletcher, J. D., & Atkinson, R. C. Evaluation of the Stanford CAI program in initial reading. *Journal of Educational Psychology*, 1972, 63, 597-602.

Fletcher, J. D., & Suppes, P. Computer-assisted instruction in mathematics and language arts for the deaf: Final report. 1973 (ERIC Document Reproduction Service No. ED 084871).

Hallworth, H. J., & Brebner, A. Computer-assisted instruction and the mentally handicapped: Some recent developments. *AEDS Proceedings*, 1978.

Jamison, D., Suppes, P., & Wells, S. The effectiveness of alternative instructional media: A survey. *Review of Educational Research*, 1974, 44, 1-67.

Knief, L. M. Effects of computer-based instruction and tutoring in an elementary school system. *AEDS Proceedings*, 1979.

Leeson, C. M., & Jaworski, J. The Hertfordshire computer-managed mathematics project. *Computers in Education*, 1975, 31-36.

Mason, G. E., & Blanchard, J. S. Computerized reading: A review. *AEDS Proceedings*, 1978.

Nelon, E. An evaluation of computer assisted vocabulary instruction with mentally retarded children. 1972 (ERIC Document Reproduction Service No. ED 090964).

Paden, D. W., Dalgaard, B. R., & Barr, M. D. A decade of computer-assisted instruction. *The Journal of Economic Education*, 1977, 9, 14-20.

- Roecks, A. L. The MICA project: A final look. AEDS Journal, 1979, 12, 146-154.
- Sevcik, E., & Sevcik, J. A learning program for problem readers. Recreational Computing, 1980, 8, 25-28.
- Suppes, P., et al. Evaluation of computer-assisted instruction in elementary mathematics for hearing-impaired students. 1973 (ERIC Document Reproduction Service No. ED 084722).
- Suppes, P., Jerman, M., & Brian, D. Computer-assisted instruction: Stanford's 1965-66 arithmetic program. New York: Academic Press, 1968.
- Thomas, D. B. The effectiveness of computer-assisted instruction in secondary schools. AEDS Journal, 1980, 12, 103-116.
- Vinsonhaler, J. F. A summary of ten major studies on CAI drill and practice. Educational Technology, 1972, 12, 29-32.
- Vitello, S. J., & Bruce, P. Computer-assisted instructional programs to facilitate mathematical learning among the handicapped. Journal of Computer-Based Instruction, 1977, 4, 26-29.

Software References

- The Math Machine. SouthWest EdPsych Services: P. O. Box 1870, Phoenix, Arizona, 1981.
- The Spelling Machine. SouthWest EdPsych Services: P. O. Box 1870, Phoenix, Arizona, 1981.

A NEW ANSWER TO THE SAME OLD PROBLEMS

Miriam Furst
Resource Teacher

Miles ELC
1400 East Broadway
Tucson, Az. 85719

Using microcomputers can be an exciting and effective way to help learning disabled and emotionally handicapped children.

Although a child with a learning disability has average or above average intellectual potential, for one or more of a variety of reasons relating to how he processes information, he does not learn effectively by standard teaching methods. Learning disabilities manifest themselves in many ways. For instance a child may handle spoken language beautifully but will be unable to process the written word. Some children may be able to decode words well but will not be able to comprehend what they read. Other children may be able to read well but won't remember the processes involved in solving simple mathematical problems.

The permutations and variations of learning disabilities sometimes seem endless. But there are some common characteristics of learning disabled children which makes the microcomputer an ideal vehicle for helping them to learn. Learning disabled children tend to be inattentive. Computers are fascinating to children and seem to compel their attention. Often, learning disabled children cannot monitor and correct their own work. The computer can be programmed to give immediate feedback and lead the child into giving correct responses. LD children have experienced failure so often that they frequently have lost the motivation to learn. Computer programs can be designed so that success is insured and reinforced.

LD children often need to have information repeated innumerable times before they can remember it. The computer can repeat itself endlessly with infinite patience.

For these reasons I decided to try using computers with the learning disabled children I work with. With the help of two high school computer science students, Paul Davis and Steve Goldberg, project TEACH was developed. TEACH is a series of computer games designed to strengthen various LD deficit areas. The majority of the games are designed so that the teacher can generate her own data files and individualize the games to the needs of each student. TEACH was developed for use on a CRT terminal interfaced with the district computer. Paul and Steve have since graduated, but other students are now in the process of gradually "translating" these games for use on the Apple microcomputer. Because of the Apple's greater graphics and sound capability we are finding that the teaching games are even more effective than they were on the CRT. A description of TEACH programs can be found at the end of this article.

In addition to using TEACH, some children in my program have been learning Basic programming. I've found that children respond very well to this. A great deal of interesting learning has taken place--the children are learning far more than just the actual mechanics of writing programs.

Teaching programming works particularly well with some of the emotionally handicapped students I work with. It took me a while to understand why programming should be such an effective technique to use with EH children. But then I began to realize that there is a common characteristic shared

by computers and many EH children. Computers, at least those that are now in common use, are by their "nature" inflexible and incapable of change. Any slight deviation on the part of the programmer from what is expected by the computer--for example a minor misspelling of a command--results in the computer's being unable to function correctly.

A characteristic of many EH children is inflexibility. They often appear to be incapable of change. At some point they seem to have chosen to follow a set of inappropriate behaviors and don't seem able to allow themselves to deviate from these behaviors. Such children have an extremely difficult time dealing with criticism. Any implication that they need to change their behavior or any attempt to have them examine and evaluate their own actions appears to be too threatening for them to deal with.

When such a child first sits down at a computer, we seem to have two immovable forces confronting each other--a computer which actually is incapable of change and a child who acts as though he is unable to change.

Even the simplest programming involves being able to find and correct errors. The computer doesn't make accommodations for the programmer's mistakes. The programmer has to find and correct them himself in order for his program to run smoothly. Implied in this process is the programmer's willingness to admit that he may have made a mistake, his willingness to search it out and the confidence that his mistakes can be corrected.

Interestingly, I've found that the motivation to produce a workable computer program is so great that once the children realize that the computer cannot change its ways, the children slowly begin to modify their own behaviors. They begin, often for the first time, to look carefully at their work and try to find mistakes rather than deny that they exist. They begin to look at mistakes in a different light. Because the process of finding and correcting mistakes allows the children's programs to function properly, this formerly distasteful experience actually becomes meaningful and exciting to the children. The children begin to develop patience and the ability to attend to detail. Their logical thinking skills begin to improve. When they show their work to others they receive a great deal of positive feedback from friends and classmates, which helps them improve their own self-images.

These positive results have been extremely encouraging. However I think it's important to mention that I use computers as only a part of the children's total program. I feel that students and teachers need to interact in a variety of different ways. But as an adjunct to a child's program I believe that the microcomputer can be an invaluable aid in teaching emotionally handicapped and learning disabled children.

"Teach" Games

1. Jimmy - visual sequestial memory

Level 1 - Child sees a word. It is removed from the screen. Child chooses correct word from four choices displayed.

Level 2 - Child sees word. Word is erased from screen and child is asked to retype word from memory.

In both levels, Jimmy is shown in jail, behind 10 bars. Each correct response removed one of the bars from Jimmy's jail. Object of the game is to set Jimmy free by getting ten correct answers. Can also be played with sequences of letters, numbers, symbols, phrases or sentences.

2. Math-visual perception, memory

Provides practice in multiplication, division, subtraction, addition or any combination of these processes. Operator can request from 1-10 digits in numbers presented and a variety of other options aimed at individualizing problems for specific needs.

3. Pattern - visual perception, position in space

Child is shown a pattern and is asked to match the correct one from four choices shown.

4. Tic Tac Toe - visual perception, reasoning

Standard tic-tac-toe game, except that children must answer a math problem before they are able to make a move.

5. High-Low - comprehension, sequencing, numerical reasoning

Child asks the computer to choose a number between 1-9, 1-99, or 1-999, and then tries to guess the mystery number. He is told whether his guesses are too high or too low.

6. Crazy Sentences - visual sequencing, reading comprehension

Child is asked to rearrange words into logical sentences.

Level 1 - Child types in numbers appearing under each word. Corresponding word then appears on screen.

Level 2 - Child types complete words in correct sequence.

7. Put Mork on Ork - visual memory, visual sequencing

Child is shown word with letters disarranged. He must type in word correctly. Each correct response moves Mork's rocket ship close to Ork.

8. See and Type - visual memory, visual attention

Child is shown a word for a designated amount of time. Word disappears and child is asked to type the word he saw.

9. Mystery Word Guess - Reading Comprehension, directionality

Child is asked to find letters above, below, to the right or left of various other letters on the keyboard. Correct responses lead to the spelling of the mystery word.

10. Beep - auditory memory, auditory attention

Child hears a random number of beeps in the time elapsed between a start and stop signal. He is asked to tell how many sounds he heard.

11. Hit and Miss - visual memory, reasoning, visual sequencing

A version of Hangman, in which child is asked to guess letters contained in a mystery word.

12. Complete the Sentence - reading comprehension

Child is asked to choose correct phrase to complete a sentence.

ALL GAMES ARE SELF CORRECTING. CORRECT AND INCORRECT RESPONSES ARE ACKNOWLEDGED. INCORRECT RESPONSES RESULT IN AN EASIER VERSION OF ORIGINAL QUESTION, REPETITION OF THE ORIGINAL QUESTION, OR A HINT LEADING TO THE CORRECT RESPONSE.

DATA FILES CAN BE MADE FOR ALL GAMES EXCEPT NUMBERS 4,5 and 9.

USING COMPUTERS WITH THE BLIND AND DEAF CHILDREN

by

Roger Goodberlet
903 N 47th Street
Phoenix, Arizona 85008

Working with handicapped children can be very challenging, which is also true of those with visual and hearing impairments. The computer has helped those with such disabilities already and posses the potential to help those individuals to lead more fulfilled independent lives. Some applications have already started while others have yet to be tested. Some of my own ideas will be introduced after a brief discussion of the current uses of the computers with the blind and deaf.

Communications with the "normal" world poses a problem for those with visual and/or hearing impairments, especially the latter. The visually impaired have braille writers, talking books, etc., but up until recently only a limited access to volumes of information. With the advent of the opticon and speech synthesizers in a computer controlled system, the blind can be read to. Almost any material can be read that is of interest to the blind. Furthermore the material can be read to the blind at their own convenience and not rely upon volunteers to read currently unavailable materials. This is crucial to students as textbooks normally must be received several months in advance of taking a class so the reader can tape the information for the student. With the computerized reader this is no longer the case. The biggest prohibitor at the moment is cost. At one time these devices ran around \$21,000 and now cost about \$5,000 for a portable reader. As the microcomputer field expands it will likely be pressed into such service at schools and homes of the blind, both adults and children. (Fast Forward)

An interesting application for the blind at the moment is the use of a date book used by a secretary in an Alaskan doctor's office. The program is not completely debussed but illustrates the potential power of speech synthesizers connected to microcomputers. (Softalk).

Communication between deaf children and deaf children and adults that know sign language is not difficult. But communications between the deaf and the hearing world is quite another matter. Here computers have been pressed into a very interesting use. By converting normal speech patterns into line configurations of the sound pattern for a visual picture of the sound spoken, the deaf person can learn to talk by matching his or her own voice pattern to the normal voice pattern for a given sound. The sounds and words are stored in computer disks and can be called up by the instructor. The student sees the visual pattern and matches it as closely as possible with that normal pattern. This very difficult process for those who teach language skills to the deaf now have an important tool on the way. The research work is currently being done at a Cambridge, Mass. University. (Fast Forward)

The computers are being used in the classroom with handicapped students of all kinds. An example is the use of 60 Apple II microcomputers at the California School for the Deaf (at the main campuses and area schools connected with CSD). From 1970 to 1972 a study was done in regards to the effectiveness of teaching deaf students through the use of microcomputers as opposed to traditional teaching methods. The study was part of a demonstration grant funded by HEW, Stanford University and CSD. Jacob Arcanin, assistant superintendent of CSD, stated the results were positive and so from 1972 through 78 use timesharing with neighboring Lawrence Hall of Science in Berkeley. With the costs and frustrations involved with the timesharing system it was recommended by Lawrence Hall that CSD go with Microcomputers. This they have done. Though some teachers are still timid, the students have embraced the microcomputers with tremendous enthusiasm. One fifth grade teacher relying on 26 years of teaching ex-

perience has this to say about computers being used with computer assisted instructional programs and the deaf. "The Apple is self teaching and builds self-confidence - even allowing students to complete independent research projects, something difficult for deaf children to master." Also, "The math drills allow the students to time themselves, graph their improvement and receive instant reinforcement. Furthermore, computer drills are far more effective than workbook drills. The students are interested in improving skills on the computer, but not in a workbook." Charlotte Beach is the teacher. (Apple Education News)

There are many uses that the computer can be put in the management of the educational program such as grading, selecting the appropriate lesson plans and of course writing the Individualized Educational Plan required under Federal Law. These educational management and administrative packages are cropping up almost daily.

Yet there are some other areas that can be developed. The biggest hindrance may be costs. One must remember when considering these costs that educational programs and providing for the living needs of the blind and deaf communities has never been anything but expensive. They must have whatever it takes to allow them to lead more independent lives. The microcomputer can be a pathway to increase that independence. With that in mind I wish to briefly list some possible uses of the microcomputer within these areas.

Currently there are some large screen terminals and magnifiers to allow the visually impaired to see what is on the screen. The blind can use braille printers which are at present cost prohibitive. Using large monitors (especially color ones) can help the visually impaired use microcomputers. Braille terminals need to be developed for the totally blind.

Greater use of large scale graphics in educational packages can assist the visually impaired in CAI programs.

Use of mass storage device to store materials read by the optical scanners and retrieved for either textual braille output or audio output through speech synthesizers. Speed of reading the optically read material into the mass storage disk drive would be the major advantage. It would have to be determined if costs could be reduced by eliminating the need for the one purpose costly optical readers currently available. Corvus is one manufacturer of the mass storage device that would be needed to store the information needed for both the read material and the speech synthesizers.

Convert braille input material via keyboard so it can be outputted to a screen to be read by a teacher more easily with contrasting colors. Trying to read dots on same color paper sometimes gets tiring. Another use is to translate the braille into regular print so the blind can send letters to their friends that to not read braille.

The reverse is true too. Have the print read back into braille or once again the audio output much as the books mentioned earlier.

Use the keyboard so the deaf can communicate with the hearing world. One such use already is the computer is connected to an emergency service number so they can relate their needs via telephone through the computer. Expansion of this idea has great potential for the deaf so they can be more able to communicate with a "normal" world.

We have not mentioned the recreational uses of the computer, especially the microcomputer. A deaf boy and his twin brother use a microcomputer to do intricate graphics work (Don't Bother Me I'm Learning TV special) about 10 years of age demonstrated the ease at which he could use a computer and the tremendous joy he had in doing this for all to see. Not only has the child something he can master, but his self confidence is greatly enhanced. Boys and girls alike can benefit from the recreational uses of the microcomputer.

Space limitations prohibit a longer discussion of the uses of microcomputers and the deaf. The list that follows are just a few places to contact if you are

interested in further information on using computers with handicapped children.

REFERENCES

Apple Education News Issue no. 2, Feb. 1980, pp 1-2; 6.

Don't Bother Me I'm Learning, a TV special presented by PBS in 1980 produced by Dave Shepardson.

Fast Forward; TV series on Computers produced in Canada and aired on KAET-TV last fall and again this spring.

Softalk; October, 1980. Sept through Nov. 1980 have articles and letters on the uses of microcomputers by the handicapped.

Those interested in CSD's program contact:

Geof Zawolkow
Coordinator, Computer Assisted Instruction
2601 Warring Street
Berkeley, CA 94704

Contact Sources
for
Programs for the Handicapped

Dr. Paul Hazan
Applied Physics Lab
John Hopkins University
John Hopkins Rd
Laurel, M20810

He has a grant from National Science Foundation to conduct a software
contest for the handicapped programs.

Gregg C. Vanderheitden, Director
Trace Research and Development Center for the Severely Handicapped
314 Waisman Center
1500 Highland Avenue
Madison Wisconsin 53706

Special area is in communication and special writing systems. Developing
custom software for the needs of handicapped individuals using the Apple
II Computer.

Softalk Assistance
10432 Burbank Blvd
North Hollywood, CA 91601

If you are handicapped or know of a handicapped person, contact for additional
help in ideas in using microcomputers to assist the handicapped.

The American Association for the Advancement of Science
Office of Opportunities in Science
1515 Massachusetts Avenue N.W.
Washington, D.C. 20005

III. GENERAL INFORMATION

341

3.1

BASIC COMPUTER BOOKS

ACCENT ON BASIC. Spencer. QA76.8 T18 R83

ADVANCED BASIC, APPLICATIONS AND PROBLEMS. Coan. QA76.8 .B3 C6

APPLE II BASIC PROGRAMMING MANUAL. Raskin. QA76.3 B3 R3

APPLESOFT II BASIC PROGRAMMING REFERENCE MANUAL. QA76.3 B3 A6

ATARI BASIC. Albrecht. QA76.8 A8 A4

BASIC. Albrecht. QA76.73 .B3 A4

BASIC: A UNIT FOR SECONDARY SCHOOLS. Spencer. QA76.73 B3 S45

BASIC AND THE PERSONAL COMPUTER. Dwyer. QA76.5 .D95

BASIC BASIC: AN INTRODUCTION TO COMPUTER PROGRAMMING IN BASIC
LANGUAGE. Coan. QA76.73 B3 C62

BASIC COMPUTER GAMES. Ahl. GV1469.2 O52

BASIC COMPUTER LANGUAGE: IT'S EASIER THAN YOU THINK! Lien.
QA76.73 B3 L54

THE BASIC COOKBOOK. Tracton. QA76.73 B3 T7

BASIC FOR BEGINNERS. Bitter and Gateley. QA76.73 .B3 G38

BASIC FOR EVERYONE. Worth. QA76.73 .B3 W67

BASIC FOR HOME COMPUTERS. Albrecht. QA76.73 .B3 A413

BASIC FROM THE GROUND UP. Simon. QA76.73 B3 S55

THE BASIC HANDBOOK, AN ENCYCLOPEDIA OF THE BASIC COMPUTER LANGUAGE.
Lien. QA76.73 B3 L53

BASIC PROGRAMMING. Kemeny. QA76.5 .K38

BASIC PROGRAMMING FOR COMPUTER LITERACY. Moursund. QA76.73 .B3 M675

BASIC PROGRAMMING PRIMER. Waite and Pardee. QA76.73 B3 W33

BASIC PROGRAMMING: SELF-TAUGHT. Hirsch. QA76.73 .B3 H58

BASIC WITH BUSINESS APPLICATIONS. Lott. HF5548.5 .B3 L67

BASIC WORKBOOK. Schoman. QA76.73 .B3 S36

BEGINNING BASIC. Chirlian. QA76.73 .B3 C54

COMPUTER ASSISTED INSTRUCTION USING BASIC. Huntington. LB1028.5 H86
 COMPUTER PROGRAMMING IN BASIC FOR EVERYONE. Dwyer. QA76.73 B3 D9
 COMPUTER PROGRAMMING IN THE BASIC LANGUAGE. Jacobs. QA76.73 .B3 C653
 COMPUTER PROGRAMMING WITH BASIC: STUDENT MANUAL. Anaheim Union High
 School District. QA76 A5
 DISCOVERING BASIC: A PROBLEM SOLVING APPROACH. Smith. QA76.5 .S618
 57 PRACTICAL PROGRAMS & GAMES IN BASIC. Tracton. QA76.6 T69
 ILLUSTRATING BASIC. Alcock. QA76.73 .B3 A42
 INSTANT FREEZE-DRIED COMPUTER PROGRAMMING IN BASIC. Brown.
 QA76.73 .B3 B75
 INTRODUCTION TO BASIC. Morton. QA76.73 .B3 M67
 INTRODUCTION TO COMPUTING WITH BASIC. Rice. QA76.73 .B3 R5
 LITTLE BOOK OF BASIC STYLE. Nevison. QA76.73 .B3 N48
 MY COMPUTER LIKES ME WHEN I SPEAK IN BASIC. QA76.73 B3 M8
 SCHAUM'S OUTLINE OF THEORY AND PROBLEMS OF PROGRAMMING WITH BASIC.
 Gottfried. QA76.73 .B3 G67
 SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS. Spencer.
 QA76.73 B3 S68
 32 BASIC PROGRAMS FOR THE PET COMPUTER. Rugg. QA76.8 .P47 R83
 32 BASIC PROGRAMS FOR THE TRS-80 (LEVEL II) COMPUTER. Rugg and Feldman.
 QA76.8 P47 R83
 VISUAL MASTERS FOR TEACHING BASIC PROGRAMMING. Spencer. QA76.73 B3 S6

Compiled by:
 Mike Zimmerman

Computer Journals
Arizona State University Microcomputer Laboratory

AEDS Journal
1201 Sixteenth St. N.W.
Washington D.C., 20036

AEDS Monitor
St. Paul Public School
360 Colborne St.
St. Paul, MN 55102

Apple ED News
10260 Bandley Drive
Cupertino, CA 95014

Byte
70 Main Street
Peterborough, NH 03458

Computer Using Educators
Mountain View High School
Mountain View, CA 94041

Creative Computing
P. O. Box 789-M
Morristown, N. J. 07960

Datamation
666 Fifth Ave.
New York, NY 10103

Education Computer Systems Group
PK3-1/M40
129 Parker Street
Maynard, MA 01754

Info World
530 Lytton Ave.
Palo Alto, CA 94301

Interface Age
McPheters, Wolfe and Jones
16704 Marquardt Ave. Box 1234
Cerritos, CA 90701

Microcomputer News
Benwill Publishing Corp.
1050 Commonwealth Ave.
Boston, MA 02215

Microcomputing
Elm Street
Peterborough, N.H. 03458

Mini-Metro Systems
Cahners Publishing Co.
221 Columbus Ave.
Boston, MA 02116

On Computing
70 Main Street
St. Peterborough, N.H. 03458

Robert Purzers Magazine
P. O. Box 466
El Dorado, CA 95623

Recreational Computing
People's Computer Co.
1263 El Camino Real, Box E
Menlo Park, CA 94025

Silicon Gulch Gazette
Computer Faire
333 Swett Road
Woodside, CA 94062

The Computing Teacher
Computing Center
Eastern Oregon State College
La Grande, OR 97850

T.H.E. Journal
Synergy, Inc.
P. O. Box 992
Acton, MA 01720

Personal Computing
1050 Commonwealth Ave.
Boston, MA 02215

Computer Related Films
Arizona State University Microcomputer Laboratory
January, 1981

	MIN	YEAR		MIN	YEAR
University of Illinois Film Library 1325 S. Oak Street Champaign, IL 81820			Modern Talking Picture Service 5000 Park Street, North St. Petersburg, FL 33709		
Basic Computer Terms	16	1976	A Few Essentials	15	1974
The Computer Revolution	22	1967	Pushing The Limits - An IBM Information Technology Report	27	1978
Computers and Human Behavior	29	1972	Flight 52	15	
Art From Computers	8	1971	About Computers	10	
A Matter of Survival	31	1969	Apex City	12	
Newspaper Production: Using Space Age Technology	16	1976	Some Call It Software	11	1972
Computers At Work	13	1970			
The Ultimate Machine	30	1972	Sandia Laboratories U.S. Department of Energy		
Communications: The Wired World	23	1972	ACCEL Revisited: Automated Circuit	20	1968
Computers: Challenging Man's Supremacy	23	1973	Computer Aided Design At Sandia Laboratories	9	1975
Computers And You	16	1972			
Computers Uses In Medicine	31	1973			
A Matter of Survival	31	1969			
			Wayne State University 5448 Cass Avenue Detroit, MI 48202		
Pacific Northwest Bell 1600 Bell Plaza Room 2802 Seattle, WA 98191			Computer Models	13	1970
Incredible Machine	15	1968			
			Farm Film Foundation 1425 H St. NW Washington, D.C. 20001		
Local Telephone Film Library			Pro AG-New Tool For Today's Farm Manager	14	
The Thinking Machines	16	1968			
I.C. A Shrinking World	18		Klein Company, Ltd., Walter J. Calculators...Classrooms... Careers	15	1978
Conquest of Light			Calculator Goes To School	29	1978
Genesis of the Transistor			How To Buy Real Estate	15	1978
Lasers Unlimited					
Lightwave	11		Gilman Engerg. 305 W. Delaware Drive Janesville, WI 53545		
Pathways					
Laser	11		Kay to Profitability: Today's World of Automated Assembly	20	1978
West Glen Films Company Of The Future, A	31	1978			
McGraw Hill Films 1221 Avenue of the Americas New York, NY 10020			Webb Company, Jervis B. Integrated Control: A New Direction in Automation	20	1974
Future Shock	45				

International Business Machines Corp.			C.S.U.F.		
1145 N. McCadden Place			Automation, The Next Revolution		
Los Angeles, CA 90038			Basic Computer Terms		
Few Essentials			Computer Crime (TV Tape) Radio		
Information Machine			& TV Dept ext. 1066		
Man and Computer..A Perspective			Introduction To Computers		
Frontiers In Computer Graphics			Weather Forecasting		
			Memory Devices (8mm)		
			Computer Simulation of the		
			Earth's Atmosphere		
			Information Processing (8mm)		
			Improving Relations Between Data		
			Processing and the user (slides)		
TRW Systems Group			Sperry Univac		
23555 Euclid Avenue			Worldwide Communications		
Cleveland, Ohio 44117			Blue Bell, PA 19425		
Best Tool Available			Computer Technology: The End-		
			less Revolution		
McDonnell Douglas Corporation			Introduction to Digital Com-		
St. Louis Library			puters		
A Better Way			Among Us		
			At The Forfront		
			The Catalyst		
Association Films, Inc.			MacMillian Films		
National Bureau of Standards			34 MacQuesten Pkwy		
Hierarchical Control: NBS			Mt. Vernon, NY 10550		
Research In Dynamic Sensors			Flow Diagram		
and Computer Control Tech-					
niques.			BFA Educational Media		
			11559 Santa Monica Blvd		
			Los Angeles, CA 90025		
			An Introduction to Computers		
			(5 filmstrips)		
Santa Fe Film Library			University of Wisconsin Film		
80 E. Jackson Blvd.			Library		
Chicago, ILL 60604			La Crosse, WI		
Argentine Yard, A Design for			Theta		
Tomorrow					
			Coronet Films		
			65 E. South Water St		
			Chicago, IL 60601		
			Data Processing: An Intro-		
			duction		
Motion Picture Bureau			University of Iowa Film Library		
Westerr Electric			Iowa City, Iowa		
222 Broadway			The Thinking Machines		
New York, NY 10038					
Micro					
Time Life Films					
100 Eisenhower Drive					
Paramus, NY 07652					
Billion Dollar Bubble					
Now The Chips Are Down					
Kent State University					
AV Services					
Kent, Ohio 44242					
Medical Electronics					
The Computer Said					
University of Arizona Film Library					
Tucson, AZ 85721					
Automation: The next Revolution					

Information Collected By:

Dan Isaacson

MICROCOMPUTER MANUFACTURERS

Compiled by Barbara Nelson

1. APF Electronics, Inc.
Imagination Machine
444 Madison Ave.
New York, N.Y. 10022
2. Apple Computer Inc.
10260 Bandley Drive
Cupertino, CA 95014
3. Atari
800 or 400
1265 Borregas Ave.
Dept. C
Sunnyvale, CA 94086
4. Commodore Pet
950 Rittenhouse Road
Norristown, PA 19401
5. Compucolor
Compucolor II Renaissance Machine
P. O. Box 569
Norcross, GA 30091
6. Cromemco Inc.
280 Bernardo Ave.
Mountain View, CA 94043
7. Exidy
Sorcerer
969 Maude Ave.
Sunnyvale, CA 94086
8. Heath Company
Heathkit
Benton Harbor, MI 49022
9. Hewlett-Packard Company
HP-85 Inquiries Mgr.
1507 Page Mill Rd.
Palo Alto, CA 94304
10. Intelligent Systems Corp.
(See Compucolor)
Intecolor Drive
225 Technology Park
Atlanta-Norcross, GA 30092
11. Mattel
Intellivision
5150 Rosecrans Ave.
Hawthorne, CA 90250
12. Motorola
5005 East McDowell
Phoenix, AZ 85008
13. North Star
Horizon II
1440-4th Street
Berkley, CA 94710
14. Ohio Scientific
Challenger Series
1333 S. Chillicothe
Aurora, OH 44202
15. Radio Shack
TRS-80
1300 One Tandy Center
Fort Worth, TX 76192
16. Sinclair Research Corp.
Sinclair Microcomputer ZX80
50 Staniford Street
Suite 800
Boston, MA 02114
17. Smoke Signal Broadcasting
31336 Via Colinas
Westlake Village, CA 91362
18. Southwest Technical Products
SWT PC 6800
219 W. Rhapsody
San Antonio, TX 78216
19. Synertek
3001 Stender Way
Santa Clara, CA 95051
20. Texas Instruments
TI 994
12501 N. Central Expressway
Dallas, TX 75222

SOFTWARE VENDORS
January 1981
Arizona State University
Microcomputer Laboratory

Apple Computer Inc.
10260 Bandley Dr.
Cupertino, CA 95014

Applecations
21650 W. Eleven Mile Rd. Suite 103
Southfield, MI 48076

Bertamax Inc.
311 W. McGraw
Seattle, WA 98119

Bluebird's Inc.
2267 23rd St.
Wyandotte, MI 48192

Blue Lakes Software
3240 University Ave.
Madison, WI 53705

Cavri Systems Inc.
26 Trumbull Street
New Haven, CT 06511

Charles Mann & Associates
7594 San Remo Trail
Yucca Valley, CA 92284

Compak Inc.
P.O. Box 14852
Austin, TX 78761

Computer Systems Design Group
3632 Govenor Dr.
San Diego, Ca 92122

Computer Systems Inter'l Inc.
539 Durie Ave.
Closter, NJ 07624

Computer Bus. Syst. of MP
1707 View St.
Myrtle Point, OR 97458

Conduit
P.O. Box 388
Iowa City, IA 52244

Cook Computer Co.
1905 Bailey Dr.
Marshalltown, IA 50158

Life Software Ltd.
146 Van Horne Ave.
Willowdale, Ontario M2J-2T1
Canada

Malibu Microcomputing
23910A De Ville Way
Malibu, CA 90265

Mean
256 N. Washington St.
Falls Church, VI 22046

Med Systems Software
P.O. Box 2674
Chapel Hill, NC 27514

Micro Gnome
5843 Montgomery Rd.
Elkridge, MD 21227

Micro Learning Ware
Box 2134
N. Mankato, MN 56001

Micro Power And Light Co.
1108 Keystone,
13773 N. Central Expwy.
Dallas, Texas 75243

Microcomputer Workshops
10 Elizabeth Place
Armonk, NY 10504

Microphys Programs
2048 Ford St.
Brooklyn, NY 11229

Monument Computer Service
Village Data Center, Box 603
Joshua Tree, CA 92284

Muse Software
330 N. Charles St.
Baltimore, MD 21201

Oregon Software
2340 SW Canyon Rd.
Portland, OR 97201

Personal Software Inc.
592 Weddell Dr.
Sunnyvale, CA 94086

Cow Bay Computing
P.O. Box 515
Manhasset, NY 11030

Creative Computing
P.O. Box 789-M
Morristown, NJ 07960

Edu-Ware Services Inc.
22035 Burbank Blvd. #223
Woodland Hills, CA 91367

Educational Activities Inc.
1937 Grand Ave.
Baldwin, NY 11510

Educational Software Prof. Ltd.
38347 Grand River
Farmington Hill, MI 48018

Educational Programs
P.O. Box 2345
W. Lafayette, IN 47906

George Earl
1302 S. Gen. McMullen
San Antonio, TX 78237

Ideatech Company
P.O. Box 62451; Dept. 7220
Sunnyvale, CA 94088

Instant Software
80 Pine St.
Peterborough, NH 03458

Interactive Education Dept. 16D
2306 Winters Dr.
Kalamazoo, MI 49002

J & S Software
140 Reid Ave.
Port Washington, NY 11050

Kyde Tyme Project; Ted Perry
2331 St. Marks Way
Sacramento, CA 95825

Level IV Products Inc.
322238 Schoolcraft; Suite F4
Livonia, MI 48154

Program Design Inc.
11 Idan Court
Greenwich, CT 06830

Programmer International Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90090

Programs for Learning Inc.
P.O. Box 954
New Milford, CT 06776

Programmer Software Exchange
2110 N. Second St., POB 199
Caobt, AR 72023

Radio Shack Dpt. Cma-81-1-161
1300 One Tandy Center
Fort Worth, TX 76102

Sliwa Enterprises
P.O. Box 7266
Hampton, Vi 23666

Steketee Educational Software
4639 Spruce St.
Philadelphia, PA 19139

The Bottom Shelf
751 Dekalb Industrial Way
Decatur, GA 30033

Thesis
P.O. Box 147
Garden City, MI 48135

TI Source
P.O. Box 191
Rye, NY 10580

TYC Software
40 Stuyvesant Manor
Geneso, NY 14454

Compiled By:

Roger Goodberlet